

Universidade Tecnológica Federal do Paraná
Departamento Acadêmico de Informática
Curso de Bacharelado em Sistemas de Informação

Melina Deraldo dos Santos

**AGENTE DE INTERFACE PARA INTERAÇÃO DE USUÁRIOS
COM O AMBIENTE VIRTUAL DE APRENDIZAGEM: MOODLE**

Trabalho de Conclusão de Curso

CURITIBA

2013

Melina Deraldo dos Santos

**AGENTE DE INTERFACE PARA INTERAÇÃO DE USUÁRIOS
COM O AMBIENTE VIRTUAL DE APRENDIZAGEM: MOODLE**

Trabalho de Conclusão de Curso apresentado à disciplina de Trabalho de Conclusão do Curso de Bacharelado em Sistemas de Informação, da Universidade Tecnológica Federal do Paraná, como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação.

Orientadora: Ana Cristina B. Kochem Vendramin

CURITIBA

2013

LISTA DE FIGURAS

Figura 1 - Aprendizagem nos Agentes de Interface	15
Figura 2 – Diagrama do Caso de Uso Conversar com o Usuário.....	36
Figura 3 – Diagrama do Caso de Uso Verificar Eventos	37
Figura 4 – Diagrama do Caso de Uso Conhecer Tutor.....	39
Figura 5 – Diagrama do Caso de Uso Verificar Comunicação entre Aluno e Tutor.....	41
Figura 6 – Diagrama do Caso de Uso Informar Funcionalidades	43
Figura 7 – Diagrama do Caso de Uso Ensinar Funcionalidades	44
Figura 8 – Modelo Entidade Relacionamento Agente de Interface.....	47
Figura 9 - Personagem Agente de Interface	49
Figura 10 - Balão de Mensagens	50
Figura 11 – Botões do Agente de Interface	50
Figura 12 - Acrescentando um novo Agente de Interface	51
Figura 13 - Tela de Boas Vindas do Agente de Interface.....	52
Figura 14 - Aviso de Eventos no Agente de Interface.....	53
Figura 15 - Informação sobre funcionalidade do MOODLE	54
Figura 16 - Funcionalidade do MOODLE no Agente de Interface	55

LISTA DE TABELAS

Tabela 1 - Propriedade de Agentes.....	13
Tabela 2 - Classificação de artigos de acordo com a abordagem de pesquisa	19
Tabela 3 - Caso de Uso Conversar com o Usuário.....	36
Tabela 4 - Caso de Uso Verificar Eventos	38
Tabela 5- Caso de Uso Conhecer Tutor	40
Tabela 6 - Caso de Uso Verificar Comunicação entre Aluno e Tutor.....	42
Tabela 7 - Caso de Uso Informar Funcionalidades	43
Tabela 8 - Caso de Uso Ensinar Funcionalidades	44
Tabela 9 - Complexidade de Casos de Uso	64
Tabela 10 - Complexidade dos Atores	64
Tabela 11 - Influência dos Fatores Técnicos e Ambientais.....	65
Tabela 12 - Fatores Técnicos.....	66
Tabela 13 - Fatores Ambientais.....	67

LISTA DE QUADROS

Quadro 1- Requisito Comunicação do Agente de Interface	31
Quadro 2 - Requisito Agente de Interface Animado	31
Quadro 3- Requisito Interface do Agente de Interface.....	32
Quadro 4 - Requisito Integrar Aplicações	32
Quadro 5 - Requisito Avisar sobre Eventos no MOODLE	33
Quadro 6 - Requisito Selecionar Tutor.....	33
Quadro 7- Requisito Comunicação entre Aluno e Tutor.....	34
Quadro 8- Requisito Ativação do Agente de Interface.....	34
Quadro 9 - Requisito Ensinar funcionalidades do MOODLE.....	34
Quadro 10- Requisito Apresentar Funcionalidades.....	35

LISTA DE SIGLAS

AVA	Ambiente Virtual de Aprendizagem
DAINF	Departamento Acadêmico de Informática
FA	Fator Ambiental
FAQ	<i>Frequently Asked Questions</i>
FTC	Fator Técnico de Complexidade
GNU	<i>GNU is Not Unix</i>
GNUGPL	<i>GNU Public License</i>
IA	Inteligência Artificial
IP	<i>Internet Protocol</i>
MOODLE	<i>Modular Object Oriented Dynamic Learning Environment</i>
MySQL	<i>My Structured Query Language</i>
PERT	<i>Program Evaluation and Review Technique</i>
PCU	Pontos de Casos de Uso
PCUNA	Pontos de Casos de Uso Não Ajustados
PHP	<i>PHP Hypertext Preprocessor</i>
SGBD	Sistema Gerenciador de Banco de Dados
UTFPR	Universidade Tecnológica Federal do Paraná
WAN	<i>Wide Area Networks</i>
WBS	<i>Work Breakdown Structure</i>
XML	<i>Extensible Markup Language</i>

SUMÁRIO

1	INTRODUÇÃO	7
1.1	MOTIVAÇÕES	8
1.2	OBJETIVO GERAL E OBJETIVOS ESPECÍFICOS	9
1.3	JUSTIFICATIVA	10
1.4	ORGANIZAÇÃO DO DOCUMENTO	10
2	AGENTES DE <i>SOFTWARE</i>	12
2.1	AGENTES DE INTERFACE	14
2.2	AMBIENTE VIRTUAL DE APRENDIZAGEM (AVA).....	16
2.2.1	MOODLE.....	17
2.3	AGENTES DE INTERFACE PARA O MOODLE	18
3	MÉTODO DE PESQUISA	22
3.1	FUNDAMENTOS	24
3.1.1	Inteligência Artificial	24
3.1.2	Personificação do Agente.....	25
3.2	TECNOLOGIAS UTILIZADAS	26
3.2.1	PHP e JQuery	26
3.2.2	SGBD: MySQL	27
3.2.3	Tecnologias para Agente.....	28
4	PROJETO DE <i>SOFTWARE</i>	30
4.1	REQUISITOS	30
4.2	DIAGRAMAS DE CASOS DE USO.....	35
4.3	ESTRUTURA DO MOODLE	45
4.4	MODELO ENTIDADE RELACIONAMENTO	46
5	GLOOB – O AGENTE DE INTERFACE.....	49
6	PROCEDIMENTOS DE TESTE E VALIDAÇÃO.....	56
7	CONCLUSÃO	58
7.1	TRABALHOS FUTUROS	59
	REFERÊNCIAS BIBLIOGRÁFICAS	60
	APÊNDICE A – ESTIMATIVA DE TEMPO PARA A CONSTRUÇÃO DO AGENTE DE INTERFACE.....	63
	APÊNDICE B – CASOS DE TESTE	69

1 INTRODUÇÃO

A ascensão de serviços baseados na Internet permite o desenvolvimento de aplicações que se caracterizam por integrar pessoas, independente da distância geográfica (FILHO, 2007). Aplicações como essas estão sendo empregadas nas mais diversas áreas, como por exemplo, comércio eletrônico, trabalho cooperativo, redes sociais, entre outras. No entanto, neste trabalho se destacam as aplicações voltadas para fins de ensino-aprendizagem.

A área de ensino-aprendizagem tem usufruído de uma ferramenta capaz de integrar alunos e professores em um ambiente virtual. Esta ferramenta pode ser utilizada para o ensino à distância ou simplesmente como apoio para o desenvolvimento de atividades de uma determinada disciplina. Ferramentas com este propósito são denominadas por Ambiente Virtual de Aprendizagem (AVA), as quais se destinam à cooperação e ao compartilhamento de conhecimento (FILHO, 2007).

Entretanto, ainda existem dificuldades por parte dos usuários para interpretar as funcionalidades de um sistema AVA e o seu modo de uso. Segundo Moraes (2005), a falta de uma interface homem-máquina adequada causa desconforto durante a interação do usuário com a aplicação AVA. Logo, prejudica a compreensão e execução das atividades disposta neste ambiente e influencia também o desempenho dos usuários (MORAES, 2005).

Para superar este desconforto durante a interação do usuário com sistemas, diversos trabalhos vem sendo desenvolvidos para aperfeiçoar a interpretação das funcionalidades das aplicações (MORAES, 2005). Segundo Braga e Pereira (2001), uma dessas abordagens, discutida dentro do campo da Inteligência Artificial (IA), é o agente de interface o qual é passível de gerar assistência para o usuário aprender a utilizar uma determinada aplicação. O agente de interface normalmente é utilizado em sistemas para auxiliar no cumprimento de tarefas que o usuário não gostaria de executar ou não compreende a execução (VAHL; OEIRAS; ROCHA, 2003).

Quando direcionados a ambientes virtuais de aprendizagem, os agentes de interface são empregados principalmente para proporcionar ao usuário a realização de

tarefas que envolvam amplo volume de informações resultante das ações executadas nestes AVAs (VAHL; OEIRAS; ROCHA, 2003). Isto é, o agente serve como um meio facilitador para a utilização do ambiente, apresentando aos usuários modos para executar as tarefas dispostas no ambiente e a interpretação das informações obtidas na relação com outros usuários.

Um agente de interface possui como característica a capacidade de ser apresentado através de um personagem familiar ao usuário. Essa personificação do agente de interface, conforme proposto por Bates (1992), traz uma “ilusão da vida”. Uma vez que o agente passa a ser representado como um personagem, ele permite que o usuário se sinta confortável ao utilizar o ambiente. Segundo Moraes (2005), isso possibilita ao usuário o prazer de interagir com um personagem real dentro de um ambiente virtual.

Sendo assim, oferecer ao usuário de ambientes AVAs a oportunidade de interagir com sistemas capazes de proporcionar um ambiente familiar é necessário para ampliar a atratividade deste tipo de ambiente.

1.1 MOTIVAÇÕES

O MOODLE é um exemplo de AVA útil à comunidade destinada ao ensino (MOODLE, 2012). Devido a esse sistema é possível proporcionar a transferência de aprendizado através de meios virtuais, principalmente à longa distância. Portanto, a apresentação deste sistema por meio da sua interface gráfica deve ser de fácil compreensão para o usuário, o qual muitas vezes não está habituado a se relacionar com sistemas como este.

É preciso vincular a esta aplicação artifícios que tragam agilidade ao seu manuseio. Sendo assim, este trabalho se propõe a criar um agente de interface que trabalhe como um guia para a utilização do MOODLE.

1.2 OBJETIVO GERAL E OBJETIVOS ESPECÍFICOS

O objetivo geral do presente trabalho é criar um método para auxiliar no processo de interação entre alunos e o AVA, denominado MOODLE (MOODLE, 2012), por meio da utilização de um agente de interface. Este agente deve ser personificado de modo a representar uma figura válida ao contexto educacional, em que os usuários estão acostumados a lidar rotineiramente.

O agente de interface deve ser capaz de identificar as principais funcionalidades do sistema e apresentar aos usuários da aplicação meios de execução de tarefas através de exemplos simplificados de manuseio. Com isso, pretende-se facilitar a visualização das principais funções do sistema perante o usuário do sistema. O agente ainda deve ser hábil para monitorar as ações realizadas pelo usuário aprendendo com as mesmas e tornando-se capaz de influenciar nas deliberações das ações a serem desempenhadas pelo usuário informando decisões tomadas anteriormente diante da mesma ação.

Em particular, o presente trabalho propõe os seguintes objetivos específicos:

- Analisar e selecionar tecnologia para desenvolver o agente de interface;
- Modelar a integração entre as duas aplicações: MOODLE e o agente de interface, provendo a interoperabilidade entre os mesmos;
- Pesquisar e compreender as funcionalidades do sistema MOODLE para interpretação do agente de interface;
- Construir a metáfora para personificar o agente de interface;
- Definir um modelo de agente animado de interface capaz de assegurar a atenção do aluno dentro do AVA;
- Modelar um método de interação entre o usuário e o agente, de forma que possibilite a aprendizagem por parte do agente animado de interface;
- Interpretar a interface do ambiente MOODLE, por meio de um agente animado de interface;
- Monitorar as ações realizadas no AVA;
- Informar ao usuário possíveis decisões a serem tomadas quanto à realização das suas atividades.

1.3 JUSTIFICATIVA

O crescimento da Internet tornou viável a utilização de ambientes virtuais para promover o processo de aprendizagem. Atualmente existem diversos aplicativos com foco no ensino, podendo ser à distância, presencial ou semipresencial. No entanto, nem sempre as funcionalidades destes ambientes são intuitivas, o que ocasiona na falta de percepção dessas funcionalidades pelos seus usuários e, muitas vezes, desinteresse dos mesmos por acreditar que as ferramentas não são capazes de suprir as suas necessidades básicas.

Por este motivo, faz-se necessária a criação de uma nova forma de interação entre o usuário e os sistemas de ensino-aprendizado. Esse novo meio de interação do sistema deve ser capaz de informar ao usuário quais as funções disponíveis na aplicação e como executá-las, se possível através de um caminho mais curto. Além disso, o ambiente deve possuir uma interface agradável à percepção do usuário. Portanto, percebe-se que a utilização de um agente de interface se aplica a tal forma de interação.

1.4 ORGANIZAÇÃO DO DOCUMENTO

O presente documento explana, no decorrer de seus capítulos, sobre a utilização de agentes de interface para facilitar a interação homem-máquina especialmente no âmbito da educação. Este primeiro capítulo tem por propósito evidenciar o objetivo deste trabalho, apresentando uma breve descrição do tema e justificando a causa para o desenvolvimento do mesmo. O segundo capítulo destina-se a conceituar o termo agente de *software* como um meio de interação entre homem e máquina. No terceiro capítulo é apresentado o método utilizado para atingir o objetivo proposto por este trabalho. O quarto capítulo apresenta o projeto de *software*. O quinto capítulo apresenta o agente de interface. O sexto capítulo aborda os procedimentos de testes e validação aplicados no agente de interface desenvolvido. Finalmente, as conclusões são fornecidas no sétimo capítulo onde são reforçados os pontos fundamentais apresentados durante todo este documento a fim de evidenciar a

importância para o desenvolvimento de um agente de interface integrado a um ambiente virtual de aprendizado com a finalidade de facilitar a compreensão da interface pelo usuário.

2 AGENTES DE *SOFTWARE*

O termo Agente é utilizado dentro da área de Inteligência Artificial (IA) (BRAGA; PEREIRA, 2001) e tem se tornado muito popular por descrever a interação entre homem e máquina. Diversas definições são empregadas para entender o amplo significado deste termo, o que tem causado inúmeros conflitos ao seu entendimento. Uma das definições para o termo agente segue:

(...) agente é um programa que em algum grau é capaz de iniciar ações, formando seus próprios objetivos, construindo planos de ações, se comunicando com outros agentes e respondendo apropriadamente aos eventos – tudo sem ser diretamente controlado por um humano. Este sentido de agente implica a existência de capacidades funcionais frequentemente relacionadas com a inteligência, adaptabilidade e responsabilidade (MORAES, 2005).

Com esta definição é possível inferir que para considerar um *software* como agente, este deve possuir uma ou mais propriedades como as dispostas na Tabela 1.

Segundo Vahl, Oeiras e Rocha (2002), cada *software* apresenta essas propriedades de acordo com o seu propósito de aplicação. No contexto deste trabalho, o termo agente é visto como um *software* autônomo capaz de agir e realizar tarefas que buscam alcançar um conjunto de metas.

Tabela 1 - Propriedade de Agentes

<i>Propriedade</i>	<i>Significado</i>
Autonomia	Agir sem intervenção humana ou de outros agentes, baseando suas ações em seu conhecimento armazenado sobre o ambiente em que atua.
Habilidade Social	Interagir com outros agentes por meio de uma linguagem comum.
Reatividade	Habilidade para sentir o ambiente e agir sobre ele.
Pró-Atividade	Não deve apenas atuar por percepção, mas deve procurar alcançar uma meta apresentando iniciativa.
Comportamento Colaborativo (Colaboração)	Trabalhar em conjunto com outros agentes a fim de atingir um objetivo comum.
Capacidade Inferencial	Agir ou abstrair a especificação de uma tarefa usando conhecimento prévio dos objetivos gerais.
Continuidade Temporal	É um processo que está continuamente sendo executado, podendo estar tanto em <i>foreground</i> ¹ quanto em <i>background</i> ² .
Personalidade	Capaz de manifestar comportamentos e propriedades de um personagem credível, tal como a emoção.
Adaptabilidade	Capaz de aprender e se aperfeiçoar com a experiência, ser capaz de reagir às mudanças do ambiente e tomar iniciativas quando apropriado.
Mobilidade	Capaz de migrar de uma máquina para outra, ou de uma plataforma para outra.

Fonte: Moraes (2005, p. 54)

Os principais tipos de agentes estão descritos a seguir (BRAGA; PEREIRA, 2001) (NWANA, 1996):

- **Agentes Colaborativos:** focam na autonomia e cooperam com outros agentes para realizar as tarefas para seus donos. São utilizados, por exemplo, em problemas muito grandes para um único agente centralizado resolver e para prover soluções para problemas inerentemente distribuídos;
- **Agentes de Interface:** este tipo de agente valoriza a autonomia e aprende para realizar tarefas em nome de seus proprietários. Interagem com o usuário através de uma das seguintes formas: observando suas ações e tentando imitá-las; recebendo um *feedback* positivo ou negativo sobre sua

¹*Foreground*: Aparecendo em primeiro plano sobre o sistema, no mesmo nível da interface gráfica.

²*Background*: O agente atua em segundo plano, juntamente com a parte lógica do sistema.

atuação no ambiente; aprendendo por meio de instruções explícitas do usuário e pedindo conselho a outros agentes;

- **Agentes Móveis:** são capazes de percorrer WANs (*WideArea Networks*) como a Web, interagindo com diferentes máquinas, capturando informações em prol de seu usuário e voltando assim que concluem as suas tarefas. Alguns benefícios provenientes do uso deste tipo de agente são a redução dos custos de comunicação, a comunicação assíncrona, a coordenação simples e o uso de uma arquitetura flexível e distribuída;
- **Agentes de Informação/Internet:** surgiram devido à demanda por ferramentas que auxiliassem no gerenciamento do crescimento explosivo de informações na Web. Realizam a gerência, manipulação e coleta de informações de várias fontes distribuídas.

De acordo com Braga e Pereira (2001), os agentes de interface, conhecidos também como assistentes pessoais, enfatizam a autonomia e o aprendizado para executarem tarefas para seus proprietários atuando quase sempre em *foreground*. Como o agente neste trabalho é classificado como um agente de interface, a próxima seção retrata maiores detalhes sobre este modelo de agente.

2.1 AGENTES DE INTERFACE

Segundo Lachi (2003), Negroponte e Kay foram os primeiros a apresentar a aplicação de agentes de interface para facilitar a interação de seres humanos com computadores, diminuindo a execução de tarefas repetitivas nos *softwares*. Desde então, técnicas vem sendo criadas e melhoradas para modelar e construir agentes de interface capazes de executar tarefas tão bem quanto um ser humano (LACHI, 2003).

Através da observação da utilização do *software*, o agente de interface obtém os padrões das ações executadas pelos usuários (RABELLO; REATEGUI; COSTI, 2007) e torna-se capaz de aprender e recomendar um modo mais eficiente de realizar uma determinada tarefa (MORAES, 2005) ou ainda automatizar a ação executada pelo usuário (RABELLO; REATEGUI; COSTI, 2007). Desta forma, o agente tem por

função principal proporcionar assistência aos usuários de modo que os mesmos consigam aprender a manusear uma determinada aplicação, fornecendo acompanhamento, aconselhamento e auxílio em contextos interativos conhecidos e ainda a serem inventados (MORAES, 2005). Segundo Nwana (1996), os agentes aprendem a melhorar o modo com que oferece apoio ao seu utilizador. Esta aprendizagem pode ocorrer de 4 (quatro) formas distintas (ver Figura 1):

1. Observando e imitando o usuário;
2. Recebendo *feedback* positivo e negativo do usuário;
3. Recebendo instruções explícitas do usuário;
4. Solicitando informação a outros agentes.

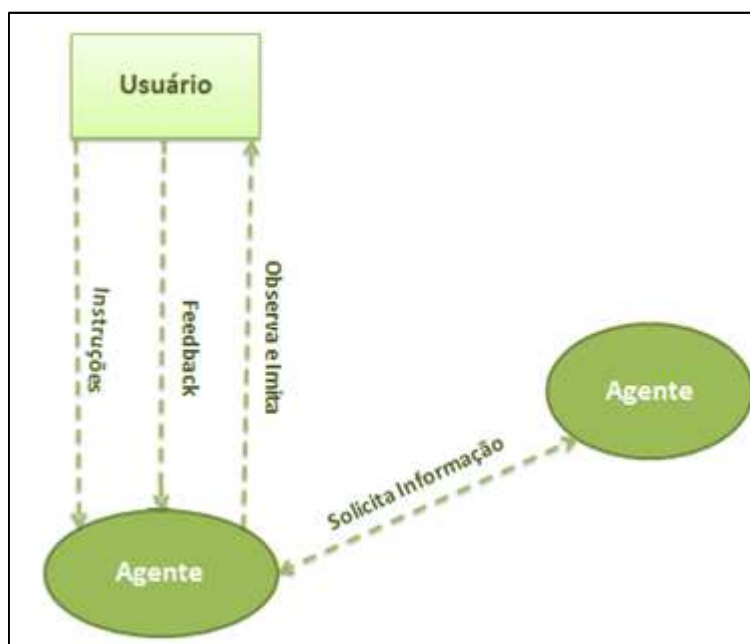


Figura 1 - Aprendizagem nos Agentes de Interface
Adaptado de Moraes (2005, p. 357)

Nas três primeiras formas o agente aprende diretamente com o usuário. Enquanto na quarta o agente utiliza o aconselhamento de outro agente, mas esta interação está limitada ao pedido de conselhos e não devem ocorrer comunicações extensas com esses parceiros como é o caso do agente da comunicação com os agentes de colaboração (MAES, 1994).

Como os agentes de interface são comumente utilizados para facilitar a interação de usuários com sistemas visando o melhor aproveitamento da aplicação pelo

usuário, a Seção 2.2 retrata um modelo de ambiente que tem se tornado popular no âmbito da educação por ser um meio facilitador de aprendizado e de comunicação entre alunos, professores, gestores e disciplinas. No entanto, ainda existem diversos estudos sobre esta aplicação a fim de constatar as influências que esta gera sobre o aprendizado.

2.2 AMBIENTE VIRTUAL DE APRENDIZAGEM (AVA)

De acordo com Filho (2007), os AVAs são ferramentas de comunicação que possibilitam aos professores, alunos e colaboradores a interação virtual com o objetivo de apoiá-los nos processos de aprendizagem. Este ambiente é útil para o desenvolvimento de condições estratégicas e intervenções de aprendizagem num escopo virtual, principalmente a WEB por utilizar a Internet como meio de comunicação.

Devido a Internet estar em período de ascensão, essa ferramenta de apoio ao aprendizado tem sido largamente utilizada e tem ganhado valor no auxílio do ensino presencial, semipresencial e principalmente no ensino à distância (CUNHA; SILVA, 2009) devido principalmente ao amplo número de funções que podem ser agregadas a ela.

As funcionalidades destes AVAs têm aumentado a cada dia, mas geralmente estas ferramentas disponibilizam meios de comunicação, de gerenciamento de conteúdo e de participantes, verificação de aprendizagem, fóruns, troca de mensagens *online*, registros de interações dos usuários, em meio a outras (CUNHA; SILVA, 2009). Ainda é possível criar um espaço individual para cada participante, considerando que o aprendizado de cada indivíduo é diferente e especializado (FILHO, 2007).

Uma plataforma para a criação e manutenção de um AVA é o MOODLE, o qual é descrito na próxima seção.

2.2.1 MOODLE

Criado em 2001 por Martin Dougiamas (educador e cientista computacional), o MOODLE (*Modular Object Oriented Dynamic Learning Environment*) é um *software* livre distribuído sob a GNU *PublicLicense*³ (GPL)(FSF, 2007) e gratuito. O MOODLE vem sendo um dos *softwares* mais utilizados pelas Instituições de Ensino Superior (IES) do Brasil (PAVEZI et al., 2011) tanto para ofertar cursos à distância ou tão somente servir de apoio para o ensino presencial.

O MOODLE disponibiliza diversos mecanismos para facilitar a interação de seus usuários, tais como (PAVEZI et al., 2011):

- Mural: ferramenta de comunicação que realiza a transmissão de informações;
- Fóruns: permite discussões *online* por meio de mensagens que são publicadas no fórum durante um determinado período de tempo;
- Lista de Discussões: utilizada para comunicação, interação e cooperação através da Internet. Por meio de conversas entre pares de indivíduos ou entre um grupo de participantes;
- Correio eletrônico: envia e recebe arquivos, anexando mensagens para emitir determinada informação;
- Enquete: coletar informações sobre um determinado assunto;
- Portfólio: conjunto de todos ou parte dos trabalhos entregue pelos usuários do AVA;
- Perfil: apresentação das informações sobre o usuário;
- FAQ (*Frequently Asked Questions*): uma lista de respostas às dúvidas mais comuns dos usuários.

Sobre aspectos técnicos, o MOODLE é uma aplicação baseada na Web formada por dois componentes: um servidor central em uma rede IP (*Internet Protocol*) (contendo os *scripts*, *softwares*, diretórios, bancos de dados, etc.) e os clientes de acesso ao seu ambiente virtual (ou seja, qualquer navegador da Web) (MOODLE, 2012).

³ Segundo a *Free Software Foundation* (2007), a GNU *Public License* permite que usuários modifiquem o código fonte dos *softwares* disponibilizados sob esta licença. No entanto, estas modificações devem ser compartilhadas com toda a comunidade usuária da aplicação, para garantir a acessibilidade da mesma.

Totalmente desenvolvido em linguagem PHP (*PHP HypertextPreprocessor*), o MOODLE suporta vários tipos de bases de dados, em especial MySQL (*My Structured Query Language*), e é idealmente implantado em servidores com o sistema operacional livre LINUX. Outra vantagem é que o MOODLE tem seu código fonte disponibilizado gratuitamente, pode ser adaptado, estendido e personalizado por quem o utiliza (MOODLE, 2012).

Por esta facilidade, o MOODLE é comumente utilizado para estudos a fim de proporcionar melhorias tanto para o ambiente quanto para a finalidade a qual se aplica: o aprendizado. A Seção 2.3 detalha mais a respeito de agentes de interface para AVAs, especialmente o MOODLE.

2.3 AGENTES DE INTERFACE PARA O MOODLE

O MOODLE foi uma ferramenta criada como uma tese de doutorado para promover um espaço de colaboração. Nesta ferramenta os usuários podiam trocar conhecimento sem a necessidade de estarem presentes no mesmo local. Atualmente, a mesma está presente em aproximadamente 198 países. No Brasil, mais de 200 instituições fazem uso deste ambiente como espaço para aprendizagem (ALVES, 2009).

Por ser uma ferramenta de código aberto e baseada na filosofia de *software* livre, existe uma comunidade motivada a interagir e modificá-la constantemente. Essa comunidade contribui para o crescimento deste ambiente e para novas customizações de acordo com diversos interesses pedagógicos. Devido a este diferencial, o MOODLE propõe a pesquisadores em educação a possibilidade de construir suas pesquisas utilizando-se de ferramentas como esta. Além disso, permite que pesquisadores estudem meios para aperfeiçoar a plataforma e sua empregabilidade.

Em vista disso, diversas pesquisas são realizadas utilizando o MOODLE como ferramenta de estudo. Dentre elas está à aplicação de agentes de interface nesta ferramenta para promover a interação do usuário com o ambiente virtual e facilitar o aprendizado.

Referindo-se à interação como a forma pela qual o usuário exerce uma relação com a ferramenta através da interface, no presente trabalho são abordadas pesquisas que relatam estudos sobre a aplicação de agentes de interface como meio facilitador para o usuário de uma ferramenta. Esses agentes são empregados tanto para aprimorar a manipulação de um ambiente quanto para melhorar a forma de transmissão do conhecimento através de AVAs, especialmente o MOODLE.

Os artigos encontrados na literatura estão de acordo com o objeto de estudo deste trabalho. Na totalidade, nove artigos referenciam a utilização de agentes de interface. Com a finalidade de agrupar os artigos de acordo com o enfoque de cada autor, a Tabela 2 apresenta o número de artigos que se enquadram nas abordagens mais relevantes e que se adequam ao tema deste trabalho. Uma das colunas apresenta o percentual dos mesmos em relação ao total de artigos encontrados.

Tabela 2 - Classificação de artigos de acordo com a abordagem de pesquisa

Abordagem de Pesquisa	Número de Artigos	Relação com o Total de Artigos (%)
Agente de Interface para diversos fins	9	100
Agente de Interface para o MOODLE	5	55,56
Agente de Interface para outros AVAs	4	44,44
Agente Animado para diversos fins	4	33,33

Fonte: Autoria Própria

Aproximadamente 56% do total de artigos encontrados apresentam estudos que utilizam o MOODLE como ferramenta para o desenvolvimento de suas pesquisas com agentes de interface. No entanto, aproximadamente 44% das pesquisas envolvem outros tipos de AVAs tais como o TelEduc, AprendEAD, que também são ambientes utilizados para o suporte da educação à distância.

Os agentes de interface criados para estes outros AVAs têm por intuito auxiliar em apenas uma das funções proporcionadas pelo AVA. Por exemplo, o agente de interface proposto por Vahl, Oeiras e Rocha (2002) contribui somente para a funcionalidade de bate-bato do TelEduc, oferecendo assistência principalmente à

recuperação de informações. No caso do AprendEAD, o agente de interface proposto por Dantas et al. (2007) tem por desígnio aparecer quando é necessário chamar atenção sobre algum procedimento, atividade ou situação, apresentando estados emocionais (representação de sentimentos similares aos de seres humanos, tais como felicidade, alegria, tristeza entre outros) para alertar o usuário.

Ao analisar os artigos que discorrem sobre os agentes de interface para o MOODLE percebe-se que 60% deles apresentam estudos de caso sobre a utilização dos agentes sobrepostos à interface deste ambiente. Desta forma, é possível verificar que somente dois artigos tem alto grau de importância para esta pesquisa. Importância esta devido ao fato de que estas pesquisas estão diretamente relacionadas ao objetivo desta proposta. Esses artigos são descritos a seguir.

O agente desenvolvido por Ribeiro, Reategui e Boff (2007) foi construído para o MOODLE e opera em *background*. Este agente nomeado por Cadinho tem por função selecionar entre um grupo de usuários do mesmo AVA um tutor para um determinado usuário. A incumbência de tutor permite que usuários inclusos na mesma disciplina do AVA, possam auxiliar uns aos outros. Este agente verifica se o usuário está cumprindo as suas atividades dentro do período proposto além de auxiliar na execução dessas atividades. O agente de interface Cadinho também é capaz de oferecer dicas de navegação sobre a interface do MOODLE, informar aos usuários as atividades recentes realizadas pelo próprio usuário ou por outros integrantes do ambiente e ainda responder perguntas realizadas em linguagem natural.

O outro agente de interface encontrado e com características similares ao agente deste trabalho é o Spyke. Este agente é desenvolvido por Ferreira e Mozzaquatro (2010) e acoplado ao MOODLE para auxiliar alunos do curso de Ciência da Computação no decorrer de um minicurso à distância. O Spyke tem por finalidade realizar bate-papos informais com usuários, isto é, responder em linguagem natural a questões realizadas por usuários. Desta forma, o agente tenta reduzir a desmotivação sentida pelo usuário devido à ausência de professores e colegas de classe no AVA.

O agente de interface criado neste trabalho, denominado Gloob, está acoplado ao MOODLE podendo ser adicionado a cada curso criado nesta ferramenta. Este agente tem por capacidade verificar o usuário que acessou o sistema e manter uma conversação através de linguagem natural com o mesmo usuário. Além disto, o agente é capaz de

informar ao usuário suas atividades e quais as funcionalidades dispostas no MOODLE e como às mesmas são executadas. Mais informações sobre o agente de interface Gloob são encontradas no Capítulo 4.

A compreensão e análise dos artigos dispostos nessa seção possuem grande relevância para a continuidade deste trabalho, uma vez que o estudo dos mesmos traz conhecimentos sobre os possíveis métodos e fundamentos a serem aplicados para completar o desenvolvimento do produto proposto. Dentre tantos métodos estudados, a escolha de um método torna-se necessária e a alternativa adotada é apresentada no Capítulo 3.

3 MÉTODO DE PESQUISA

Um agente de interface, segundo Moraes (2005), deve ser competente no domínio da aplicação ou no ambiente em que opera. Com o intuito de se tornar competente este deve possuir acesso a todas as informações e operações possíveis no ambiente em que está localizado. E ainda, um agente de interface pode produzir meta-conhecimento⁴ e a múltipla representação⁵ do mesmo. Também deve ser acessível, isto é, o usuário pode predizer o que o agente pode fazer em determinada situação. Por este motivo, a representação externa do agente deve estar de acordo com a natureza da tarefa a ser realizada e com as características do ambiente no qual está inserido (MORAES, 2005).

Para alcançar uma representação externa do agente de forma que este fosse facilmente compreendido pelo usuário, modelos de representação de interfaces ou personificação foram estudados. Estes modelos permitem que o *software* pareça familiar e conseqüentemente fácil de manipular (MORAES, 2005). Desta forma, o agente se torna competente para auxiliar o usuário e entretê-lo durante o período de execução de uma determinada tarefa.

Após definir a personificação do agente de interface, foi necessário determinar como a interação do agente com o usuário seria realizada, para isto foi importante verificar a aceitação dos usuários sobre a automatização das ações propostas pelo agente. A automatização das ações deve se enquadrar aos desejos do usuário. Isto é, deve-se permitir ao usuário o direito de escolha sobre a execução da ação fornecida pelo agente. É preciso deixar claro que o processo de automatização é executado permitindo que o usuário também possa executar ações sem a influência do agente (Moraes, 2005).

Portanto, foi preciso decidir qual o método de controle que o usuário teria sobre o agente. Este domínio foi estudado para que o agente não se tornasse o controlador do ambiente, fazendo com que o usuário perdesse o controle e dependesse integralmente do agente de interface para realizar suas tarefas. Esse método de controle

⁴ Entende-se por meta-conhecimento, o conhecimento para solucionar um determinado problema em um domínio.

⁵ Múltipla representação é a habilidade de produzir representações alternativas de informações conforme a necessidade e estilo de um usuário.

foi implantado utilizando um espécie de *feedback* fornecido pelo usuário. Assim, o agente de interface ainda conseguiria aprender sobre o usuário.

Como o agente precisa aprender sobre o AVA e ainda deliberar as ações executadas neste ambiente, o agente foi desenvolvido em linguagem compatível com a utilizada no desenvolvimento do AVA. Além disso, foi preciso utilizar meios propícios para a comunicação entre as duas aplicações. Estes meios foram estudados e analisados de forma que fosse escolhido o que mais se adequasse ao MOODLE e que acarretasse em maior número de benefícios.

O agente de interface também precisava se adequar ao AVA para interpretar as informações obtidas a partir da base de dados do AVA, mantida por um Sistema Gerenciador de Banco de Dados (SGBD). Desta forma, a base de dados do agente foi criada dentro no mesmo SGBD do AVA, o que facilitou a transferência de informação entre o agente de interface e o ambiente. Com a definição da base de dados foi possível inserir a base de conhecimento inicial do agente de interface. Esta base de conhecimento foi definida utilizando-se mecanismos propostos pela Inteligência Artificial, a qual é apresentada na Seção 3.1.1.

Outro propósito do agente de interface, que também utiliza de conceitos e técnicas da IA, é o aprendizado do mesmo. Isto ocorre de acordo com os dados obtidos sobre o usuário, os eventos gerados pelo usuário e as funcionalidades dispostas no MOODLE. A interpretação destes fatos possibilita ao agente de interface apresentar ao usuário possíveis ações a serem realizadas de acordo com os itens dispostos no AVA.

Logo em seguida, o agente foi implantado em um ambiente real para a integração com o MOODLE de modo que fosse possível a realização de testes de integração das aplicações. Isso permitiu a verificação de possíveis falhas e a correção das mesmas.

O AVA provido pela ferramenta MOODLE é utilizado para a integração com o agente e é fornecido pelo Departamento Acadêmico de Informática (DAINF) da Universidade Tecnológica Federal do Paraná (UTFPR). Para esta concessão é necessário que se assine um termo de responsabilidade sobre os dados ali expostos. Este termo é necessário, pois, a base de dados proporciona informações de usuários reais do sistema. A utilização de uma base de dados contendo informações reais sobre os

usuários do MOODLE permite que o agente aprenda com as ações que foram realmente realizadas.

O agente de interface proposto neste trabalho necessita de alguns fundamentos para cumprir as suas funcionalidades, os quais são descritos na sequência.

3.1 FUNDAMENTOS

Nesta seção, os principais fundamentos necessários para a construção do agente de interface são apresentados. Iniciando com o método pelo qual o agente construiu o seu aprendizado e realizar a comunicação com o usuário da aplicação. Estas duas funcionalidades estão englobadas no âmbito da inteligência artificial que está descrita a seguir.

3.1.1 Inteligência Artificial

Como previamente descrito no Capítulo 2, o conceito de agente de interface está diretamente vinculado à área de Inteligência Artificial. Esta é uma área da computação que possui diversas definições devido ao seu vasto campo de atuação. A definição cabível a esta pesquisa está relacionada com a estratégia de modelagem cognitiva. Isto é, o agente proposto além de influenciar na maneira de aprendizado do usuário também deve ser capaz de aprender sobre as ações humanas tomadas no ambiente em que está inserido (RUSSEL; NORVIG, 2004).

O agente de interface desenvolvido é racional, isto é, o agente é capaz de escolher os meios mais viáveis para atingir o sucesso de suas ações, sucesso este que é descrito no processo de análise e projeto do agente. Outra característica de um agente racional desenvolvido neste trabalho é a necessidade de aprender sobre o usuário para servir de meio facilitador. De acordo com as percepções que o agente tem do ambiente e as experiências vivenciadas, o agente pode modificar e/ou ampliar os conhecimentos

iniciais (RUSSEL; NORVIG, 2004). Os conhecimentos iniciais são inseridos pelo construtor somente para que o agente entenda o ambiente no qual foi implantado. A aquisição posterior de conhecimento pelo próprio agente assegura a sua autonomia.

Quanto mais o agente aprende sobre o ambiente em que está inserido, mais ele elimina os conhecimentos parciais ou incorretos anteriormente definidos. Por isso, é necessário fornecer ao agente um conhecimento inicial, principalmente sobre a forma como o mesmo realiza o aprendizado no ambiente. Isso permite que o agente se torne efetivamente independente dos conhecimentos iniciais.

O agente de interface produzido neste trabalho possui uma relação de perguntas, que são realizadas ao usuário, armazenada em sua base de dados. Além destas informações, o agente possui como conhecimento inicial as informações contidas no MOODLE sobre o usuário, curso e o próprio sistema. Essas informações são importantes para que o agente conheça todo o ambiente em que está inserido além de conhecer o usuário e desta forma conseguir auxiliá-lo.

Os algoritmos dispostos pela IA que oferecem racionalidade ao agente foram estudados durante o andamento deste trabalho, no sentido de escolher o mais apropriado para o aprendizado das ações do usuário perante o MOODLE. Além de garantir base para a autonomia do agente conforme as suas percepções do ambiente.

Outro fator importante do agente é garantir que o mesmo possua uma interface aceitável pelo usuário da aplicação. A fim de garantir este consentimento do usuário, um modelo de personagem para o agente é proposto na próxima seção.

3.1.2 Personificação do Agente

O conceito de se utilizar personificação surgiu durante a constatação de que as pessoas interagem com personagens em todos os aspectos da sua vida, sendo estes fictícios ou reais. Personagens fictícios nos transportam para suas histórias nos filmes, livros e TV. Na realidade, fazemos parte de um conjunto de personagens reais que são nossa família, amigos e colegas de trabalho (MORAES, 2005).

No entanto, habilidades sociais de interação como a fala e a movimentação, não são utilizadas para gerenciar e obter informação nas aplicações de computadores. Ultimamente, os usuários tem que habituar-se aos modelos já estabelecidos. Contudo, isto não deveria ocorrer, pois de acordo com Elliot e Bezezinsky (1998) as interfaces devem se adequar ao usuário. O progresso das interfaces está em reduzir as necessidades de adequação dos usuários e aumentar a habilidade da aplicação de se comunicar com o mesmo.

Fez-se então necessário definir de maneira clara a personagem para representar o agente de interface. Assim sendo, foi importante explorar o ambiente em que o agente é inserido, principalmente, captar do atual usuário qual o modelo ou objeto de interação que ele está mais habituado com relação ao seu aprendizado. Incluir o usuário no momento de escolha, ou conhecer o tipo do usuário a ponto de conseguir definir a qual personagem o agente de interface deve se enquadrar, exige cuidado, pois o agente deve criar um vínculo familiar com o usuário e é cobrado como tal (MORAES, 2005). Assim, se o agente não apresentar as funções esperadas pelo usuário, este é considerado “incompetente”.

As tecnologias utilizadas para o desenvolvimento do agente são descritas na Seção 3.2.

3.2 TECNOLOGIAS UTILIZADAS

Considerando que o MOODLE está pronto e utiliza ferramentas próprias para o seu desenvolvimento, o agente deve utilizar as mesmas tecnologias ou similares. As seções seguintes descrevem as tecnologias necessárias para o desenvolvimento do MOODLE.

3.2.1 PHP e JQuery

O ambiente MOODLE é inteiramente desenvolvido em PHP (MOODLE, 2012) que é uma linguagem de código aberto e multi-plataforma (pode ser executado

sobre plataformas Unix, Linux, Windows, entre outras). Devido à necessidade de interoperabilidade entre o agente de interface e o AVA, o agente de interface deve ser desenvolvido na mesma linguagem do AVA.

Entretanto, o agente de interface deve ser dinâmico durante a sua comunicação com o usuário. Por isto, a biblioteca jQuery foi integrada ao código em PHP. Esta biblioteca é útil para a manipulação de eventos e animação de forma simples. A combinação destas duas linguagens torna possível apresentar o agente de interface de forma dinâmica ao usuário.

Com essas duas tecnologias o agente pode se comunicar com o usuário por meio de balões de mensagem. Esses balões são atualizados dinamicamente, segundo as respostas que o usuário fornece ao agente.

A fim de manipular os dados que transitam no MOODLE, a seção a seguir apresenta o banco de dados utilizado para persistir os dados do MOODLE.

3.2.2 SGBD: MySQL

O MySQL é um SGBD de código aberto que suporta praticamente todas as plataformas atuais e é compatível com diversas linguagens de programação, disponibilizando *drivers* e módulos de interface para comunicação entre elas (MySQL, 2012). O MySQL um *software* livre com base na GPL.

Atualmente, o MySQL é a ferramenta para manipulação de dados utilizada pelo MOODLE. Esta também é utilizada para construir a base de dados do agente de interface. A escolha da mesma ferramenta para a manipulação de dados é importante para manter a compatibilidade entre o agente de interface e o MOODLE.

Além disto, para manter o agente de interface compatível com o MOODLE é necessário definir um personagem que interprete a funcionalidade desta aplicação. A seção a seguir discute sobre as tecnologias utilizadas para a personificação do agente de interface.

3.2.3 Tecnologias para Agente

Diversas tecnologias para a personificação de agentes de interface foram descritas nos artigos estudados durante o levantamento bibliográfico descrito no Capítulo 2. Dentre estas tecnologias se destacaram o *OddcastVhost* (FERREIRA; MOZZAQUATRO 2010) e a *Microsoft Agent*.

A *OddcastVhost* é uma tecnologia proprietária que permite a criação de personagens interativos com base em linguagem Flash (linguagem de programação voltada para web). Para ter acesso a esta tecnologia é necessário realizar a compra do *software* proprietário. Sendo assim, a possibilidade de utilização da *OddcastVhost* foi descartada.

A *Microsoft Agent* é uma ferramenta comercializada pela *Microsoft* que contém agentes de interface já prontos e que podem ser implantados dentro de um ambiente que servem para facilitar a construção de agentes próprios. No entanto essa tecnologia foi descontinuada pela *Microsoft* desde o *Windows 7* e a *Microsoft* (2013) desaconselha o seu uso devido à falta de manutenção. Devido à descontinuidade, esta tecnologia também foi descartada para o uso neste projeto.

Com o descarte das tecnologias apresentadas, considerou-se o uso da linguagem PHP para o desenvolvimento do agente de interface, devido ao fato de que o sistema no qual o agente será inserido é desenvolvido totalmente em PHP e a documentação para o desenvolvimento de novos módulos pede para que ao inserir um novo módulo o mesmo seja criado na linguagem do sistema de origem (MOODLE Docs, 2013). Desta forma, ainda é possível evitar prováveis problemas com a integração de linguagens. Seguindo a estrutura do MOODLE (ver Seção 4.3) e suas dependências foi criado um arquivo de *lib.php* para armazenar as funções em PHP que seriam pertinentes ao agente de interface. Essas funções foram criadas para executar as ações do agente de interface perante o usuário. Além disto, as funções tem por característica acessar os dados do usuário que estão persistidos na base de dados do ambiente MOODLE e com esses dados o agente é capaz de realizar o processamento e apresentar informações aos usuários por meio dos seus balões de mensagens.

Além disto, a biblioteca jQuery é aplicada para realizar atualizações na tela do agente de interface de forma que seja imperceptível ao usuário. Isto é, cada vez que o agente de interface precisar atualizar algum componente de sua tela (ex: balões de mensagens e botões) não é necessário recarregar toda a página. Essa decisão foi importante para evitar lentidão no sistema e transtorno no uso do agente de interface. Para o usuário realizar *feedback* ao agente de interface foram disponibilizados botões na tela do agente de interface. Esses botões são disponibilizados de acordo com a necessidade da resposta da mensagem apresentada. Sendo as respostas das mensagens limitadas em ‘Sim’ ou ‘Não’. No entanto, o agente ainda apresenta mensagens que são apenas informações, necessitando apenas de um botão ‘Próximo’ a fim de seguir para o próximo passo.

A cada *feedback* que o usuário fornece ao agente de interface por meio dos botões, a resposta é armazenada na base de dados. Essa resposta é armazenada para posteriormente o agente de interface melhorar a sua compreensão sobre o usuário. E assim, aprimorar a forma com que interage com o usuário do MOODLE.

Após discutir sobre as tecnologias que serão aplicadas para o desenvolvimento do agente de interface, o Capítulo 4 retrata o projeto de *software* necessário para a produção do agente de interface.

4 PROJETO DE SOFTWARE

Para produzir uma solução coerente ao desenvolvimento do agente de interface, este capítulo se destina a definir os detalhes para a construção do agente. Sendo assim, torna-se necessário produzir um modelo que represente o agente identificando o que é preciso para produzir para o desenvolvimento do mesmo.

A definição deste modelo ocorre a partir do levantamento de requisitos, o qual determina as particularidades do agente.

4.1 REQUISITOS

Os requisitos do agente de interface são a descrição das funcionalidades que o agente deve fornecer e as suas restrições operacionais. Desta forma, os requisitos retratados nesta seção definem exatamente o que está sendo desenvolvido.

Os requisitos apresentados são classificados em funcionais ou não funcionais. Os requisitos funcionais definem as funcionalidades que o sistema deve oferecer e como o sistema deve reagir em uma determinada situação. Enquanto isso, os requisitos não funcionais apresentam as restrições sobre algum serviço.

Nos quadros a seguir estão dispostos os requisitos necessários para o desenvolvimento do agente de interface. Cada requisito possui um código de identificação que é definido por RXXX, sendo que o termo R representa um requisito e o X representa a identificação do requisito através de um número sequencial. Desta forma, o primeiro requisito é identificado por R001. Além disso, o quadro de representação de cada requisito possui o tipo em que o requisito se enquadra (funcional ou não funcional), o evento que o representa (isto é, o código do caso de uso que formaliza o requisito), uma breve descrição sobre o mesmo e se ele depende de algum outro requisito.

O Quadro 1 apresenta o requisito referente à comunicação entre o usuário do MOODLE e o agente de interface.

Requisito < Comunicação do Agente de Interface>			
Código:	R001	Tipo FUNCIONAL	Evento: UC001
Descrição:	O agente de interface deve utilizar linguagem natural para se comunicar com o usuário do MOODLE. Esta linguagem deve ser especificada de acordo com o perfil do usuário.		
Dependência:	Não se Aplica		

Quadro 1- Requisito Comunicação do Agente de Interface

Fonte: Autoria Própria

O Quadro 2 apresenta o requisito que menciona sobre a animação do agente a fim de aumentar a interação com usuário.

Requisito < Agente de Interface Animado>			
Código:	R002	Tipo FUNCIONAL	Evento: UC006
Descrição:	O agente de interface deve apresentar alguma forma de animação ao se comunicar com o usuário do MOODLE.		
Dependência:	Não se Aplica		

Quadro 2 - Requisito Agente de Interface Animado

Fonte: Autoria Própria

O Quadro 3 expõe uma interface de fácil compreensão para o agente de interface, além disso, a mesma deve ser adequada ao perfil do usuário.

Requisito < Interface Agente de Interface>			
Código	R003	Tipo FUNCIONAL	Evento: UC006
Descrição:	O agente de interface deve se apresentar de forma amigável. Isto é, através de um personagem que seja adequado ao perfil dos usuários do MOODLE.		
Dependência:	R002		

Quadro 3- Requisito Interface do Agente de Interface

Fonte: Autoria Própria

O Quadro 4 refere-se à integração que o agente de interface deve ter com o MOODLE. Esta deve ser bem definida para que não ocorram erros de comunicação entre as aplicações.

Requisito < Integrar Aplicações>			
Código	R004	Tipo FUNCIONAL	Evento: UC004
Descrição:	O agente de interface deve ser integrado ao MOODLE, gerando interoperabilidade entre eles.		
Dependência:	Não se Aplica		

Quadro 4 - Requisito Integrar Aplicações

Fonte: Autoria Própria

O requisito R005, disposto no Quadro 5, deseja que o agente avise ao usuário sobre os novos eventos que foram adicionados ao sistema. Entre esses eventos estão: novas tarefas, criação de fóruns e *wikis*, avisos e adição de novos materiais na disciplina.

Requisito < Avisar sobre Eventos no MOODLE>	
Código	R005 Tipo FUNCIONAL Evento: UC002
Descrição:	O agente de interface deve informar aos usuários os eventos abertos no MOODLE.
Dependência:	Não se Aplica

Quadro 5 - Requisito Avisar sobre Eventos no MOODLE

Fonte: Autoria Própria

A seleção de um colega de disciplina para auxiliar um usuário em alguma atividade do MOODLE é o requisito descrito no Quadro 6. Este requisito especifica a seleção que o agente deve fazer entre os alunos de uma determina disciplina para atuar como auxiliador de outro aluno. O usuário tem a opção de negar a seleção do tutor feita pelo agente de interface. Neste caso o agente de interface deve selecionar outro aluno para atuar como tutor do usuário.

Requisito < Selecionar Tutor>	
Código	R006 Tipo FUNCIONAL Evento: UC003
Descrição:	O agente de interface deve selecionar um aluno para realizar atividades de tutoria que auxilie outro aluno com dificuldades na mesma disciplina.
Dependência:	Não se Aplica

Quadro 6 - Requisito Selecionar Tutor

Fonte: Autoria Própria

A comunicação entre o aluno e o seu tutor precisa ser sugerida pelo agente de interface conforme descrito no requisito do Quadro 7. Sendo assim, o agente deve sugerir ao aluno formas dentro do MOODLE de realizar a comunicação com seu tutor até o término das atividades propostas. No entanto, o aluno pode optar por utilizar outros meios de comunicação fora do ambiente MOODLE. Neste caso, o agente não poderá verificar a comunicação entre o aluno e seu tutor. Então, é dever do aluno avisar ao agente que a comunicação está ocorrendo fora do ambiente MOODLE.

Requisito < Comunicação entre Aluno e Tutor >			
Código	R007	Tipo FUNCIONAL	Evento: UC004
Descrição:	O agente de interface deve informar meios de comunicação entre aluno e tutor, o agente deve sugerir o bate-papo para troca de informações.		
Dependência:	R006		

Quadro 7- Requisito Comunicação entre Aluno e Tutor

Fonte: Autoria Própria

O agente de interface pode ser ativado ou desativado pelo usuário a qualquer momento, conforme exposto no requisito do Quadro 8.

Requisito <Ativação do Agente de Interface >			
Código	R008	Tipo FUNCIONAL	Evento: UC006
Descrição:	O agente de interface poderá ser ativado pelo usuário sempre que este desejar. Caso o usuário prefira manter o agente oculto, o mesmo deve permanecer desativado até o usuário decidir liberar a execução do agente novamente.		
Dependência:	Não se Aplica		

Quadro 8- Requisito Ativação do Agente de Interface

Fonte: Autoria Própria

O Quadro 9 apresenta o requisito que indica a necessidade do agente ensinar o usuário a utilizar as funcionalidades dispostas no MOODLE, as quais foram descritas na Seção 2.2.1.

Requisito < Ensinar funcionalidades do MOODLE >			
Código	R009	Tipo FUNCIONAL	Evento: UC005
Descrição:	O agente de interface deve ensinar o usuário a executar as funcionalidades dispostas no MOODLE.		
Dependência:	Não se Aplica		

Quadro 9 - Requisito Ensinar funcionalidades do MOODLE

Fonte: Autoria Própria

O último requisito, descrito no Quadro 10, inclui a apresentação das funcionalidades do MOODLE pelo agente de interface. Desta forma, o usuário ficará ciente de todas as opções disponíveis no MOODLE. E caso necessite saber como utilizá-las, o requisito R009 (ver Quadro 9) deve ser aplicado.

Requisito < Apresentar Funcionalidades >			
Código	R010	Tipo FUNCIONAL	Evento: UC005
Descrição:	O agente de interface deve apresentar as funcionalidades disponíveis no MOODLE do usuário.		
Dependência:	Não se Aplica		

Quadro 10- Requisito Apresentar Funcionalidades

Fonte: Autoria Própria

Para formalizar os requisitos a seção seguinte apresenta os diagramas de Casos de Uso do agente.

4.2 DIAGRAMAS DE CASOS DE USO

Os diagramas de Casos de Uso formalizam os requisitos levantados para o desenvolvimento do agente de interface. Eles são usados para possibilitar a verificação de possíveis problemas e ambiguidades nos requisitos (SOMMERVILLE, 2007).

Cada caso de uso apresentado demonstra um modo de interação entre o agente de interface e o usuário do MOODLE. Os mesmos são identificados por um código o qual é composto por UC, uma abreviatura para *Use Cases* (casos de uso), mais uma sequência numeral crescente. Desta forma, o primeiro caso de uso é identificado por UC001.

O caso de uso apresentado na Figura 2 apresenta a comunicação entre o agente de interface e o usuário do sistema MOODLE. Para que a comunicação ocorra de maneira eficaz ele necessita integrar todos os requisitos que consideram além da comunicação a interface utilizada pelo agente.



Figura 2 – Diagrama do Caso de Uso Conversar com o Usuário
Fonte: Autoria Própria

A Tabela 3 apresenta a descrição do caso de uso Conversar com o usuário, o qual é responsável pela interação do usuário com o agente, através de uma linguagem formal. Além disso, a Tabela 3 informa os passos necessários para que o usuário consiga realizar a ação proposta neste caso de uso.

Tabela 3 - Caso de Uso Conversar com o Usuário

Nome	UC001 – Conversar com Usuário
Descrição	O caso de uso UC001 é responsável por realizar a comunicação do agente de interface com o usuário do MOODLE
Ator Principal	Agente
Ator de Bastidor	Usuário
Pré Condições	O agente deve estar ativo
Pós Condições	
Fluxo Básico	1. O usuário entra na página inicial do MOODLE. 2. O agente inicia o diálogo com o usuário. 3. O usuário responde o agente.
Regras de Negócio	R001 e R004

Fonte: Autoria Própria

A Figura 3 retrata o caso de uso UC002, o qual é responsável por definir como o agente de interface apresenta os eventos criados nas disciplinas para o usuário.



Figura 3 – Diagrama do Caso de Uso Verificar Eventos
Fonte: Autoria Própria

A Tabela 4 apresenta uma breve descrição de como o caso de uso Verificar Eventos, ocorre dentro do ambiente MOODLE, as condições e os passos necessários para executá-lo e os requisitos responsáveis para a sua definição.

Tabela 4 - Caso de Uso Verificar Eventos

Nome	UC002 – Verificar Eventos
Descrição	Este caso de uso é responsável por identificar os eventos e notificações criados nas disciplinas do MOODLE e informar ao usuário
Ator Principal	Agente
Ator de Bastidor	Usuário
Pré Condições	O agente deve estar ativo. Um novo evento ou notificação foi criado.
Pós Condições	O usuário foi informado sobre o novo evento.
Fluxo Básico	<ol style="list-style-type: none"> 1. O agente verifica novos eventos criados no MOODLE. (A1) 2. O agente informa sobre os novos eventos ao usuário 3. O agente pergunta se o usuário quer ir para a página do evento. 4. O usuário confirma. (A2) 5. O agente encaminha o usuário para a página do evento.
Fluxo Alternativo	<p>A1 Usuário pergunta ao agente</p> <ol style="list-style-type: none"> 1. O aluno pergunta ao agente sobre os eventos 2. O agente verifica os eventos no MOODLE 3. O agente informa ao usuário sobre os eventos que pertencem a sua disciplina. 4. Retorna ao passo 3 do fluxo Básico <p>A2 Cancelar</p> <ol style="list-style-type: none"> 1. O usuário cancela a ação do agente.
Regras de Negócio	R005

Fonte: Autoria Própria

O caso de uso denominado por Conhecer Tutor é apresentado na Figura 4. Este caso de uso define a maneira como o agente de interface seleciona um tutor para um

determinado usuário do sistema. Para isto o usuário deve pedir ao agente que o mesmo informe qual o tutor selecionado para ele.

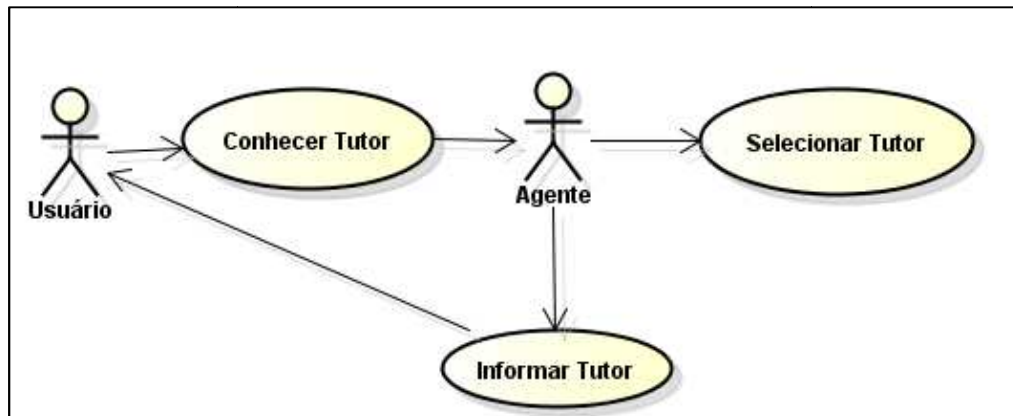


Figura 4 – Diagrama do Caso de Uso Conhecer Tutor
Fonte: Autoria Própria

A Tabela 5 apresenta informações de como executar este caso de uso conhecer tutor por meio de um fluxo principal e os atores responsáveis por gerar tal evento.

Tabela 5 - Caso de Uso Conhecer Tutor

Nome	UC003 - Conhecer Tutor
Descrição	Este caso de uso é responsável por apresentar ao aluno um Tutor (também usuário da mesma disciplina) para acompanhá-lo na execução de suas tarefas. O agente de interface deve selecionar um aluno de forma aleatória para exercer as atividades de tutor e apresentá-lo ao aluno que é tutorado. Além disso, quem foi selecionado para ser tutor deve ser informado pelo agente sobre a sua nova função.
Ator Principal	Agente
Ator de Bastidor	Usuário
Pré Condições	O agente deve estar ativo. O usuário deve desejar conhecer seu tutor.
Pós Condições	<ol style="list-style-type: none"> 1. O usuário deve ser informado sobre o seu tutor. 2. O aluno tutor deve ser informado que irá auxiliar um determinado usuário;
Fluxo Básico	<ol style="list-style-type: none"> 1. O usuário pede para conhecer o seu tutor. 2. O agente seleciona um aluno da mesma disciplina para ser o tutor. 3. O agente informa ao usuário o nome do seu novo tutor. 4. O agente informa ao tutor a sua nova atividade.
Regras de Negócio	R006

Fonte: Autoria Própria

O caso de uso da Figura 5 verifica se está ocorrendo comunicação entre o aluno e o seu tutor através do bate-papo do MOODLE.

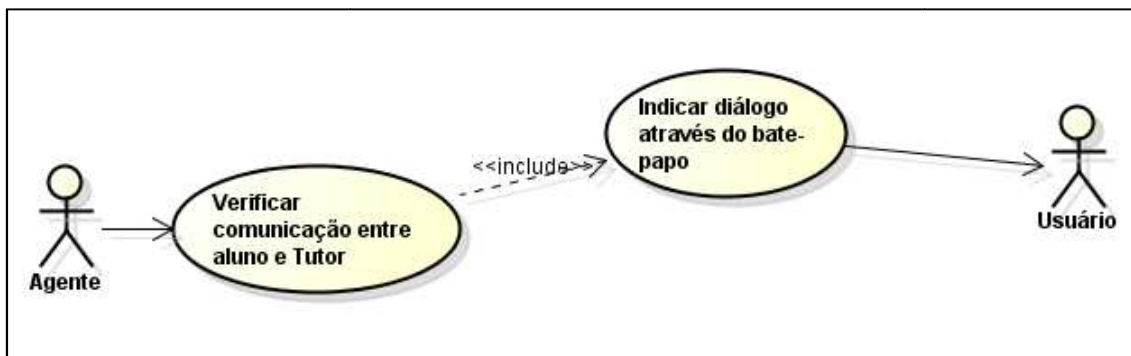


Figura 5 – Diagrama do Caso de Uso Verificar Comunicação entre Aluno e Tutor
Fonte: Autoria Própria

Conforme a Tabela 6 o caso de uso Verificar Comunicação entre Aluno e Tutor é o evento gerado pelo requisito R007. Este caso de uso deve verificar se o aluno e o seu tutor estão trocando informações sobre as atividades das disciplinas do MOODLE.

Tabela 6 - Caso de Uso Verificar Comunicação entre Aluno e Tutor

Nome	UC004 - Verificar comunicação entre aluno e tutor
Descrição	Este caso de uso é responsável por verificar se o aluno e o tutor estão trocando informações pelo MOODLE.
Ator Principal	Agente
Ator de Bastidor	Usuário
Pré Condições	O agente deve estar ativo. O usuário deve possuir um tutor.
Pós Condições	O usuário foi informado sobre o novo evento.
Fluxo Básico	<ol style="list-style-type: none"> 1. O agente verifica a troca de informações entre tutor e usuário no MOODLE. 2. O agente informa ao usuário que não está ocorrendo troca de informação com o seu tutor. 3. O agente sugere o uso do bate-papo para a troca de informações. 4. O usuário aceita a sugestão do agente. (A1) 5. O agente encaminha o usuário para o bate-papo
Fluxo Alternativo	<p>A1 Avisar Comunicação fora do MOODLE</p> <ol style="list-style-type: none"> 1. O usuário informa ao agente que esta trocando informações fora do MOODLE.
Regras de Negócio	R007

Fonte: Autoria Própria

O caso de uso apresentado na Figura 6 retrata o momento em que o agente deve informar ao usuário as funcionalidades disponíveis no MOODLE. Este caso de uso poderá ser ativado pelo usuário quando ele quiser saber mais sobre as funcionalidades do AVA.

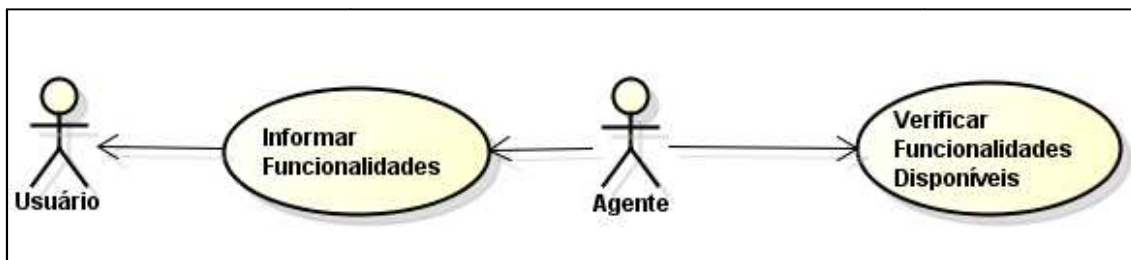


Figura 6 – Diagrama do Caso de Uso Informar Funcionalidades
Fonte: Autorial Própria

Tabela 7 descreve a função do caso de uso Informar Funcionalidades apontando os atores envolvidos e os passos que devem ser seguidos para executar o mesmo. Além disso, apresenta qual o requisito que gera este caso de uso.

Tabela 7 - Caso de Uso Informar Funcionalidades

Nome	UC005 - Informar Funcionalidades
Descrição	Este caso de uso é responsável por informar ao usuário as funcionalidades disponíveis.
Ator Principal	Agente
Ator de Bastidor	Usuário
Pré Condições	O agente deve estar ativo.
Fluxo Básico	<ol style="list-style-type: none"> 1. O usuário pergunta ao agente sobre o MOODLE. (A1) 2. O agente informa ao aluno sobre o MOODLE. 3. O agente informa sobre cada funcionalidade
Regras de Negócio	R009 e R010

Fonte: Autorial Própria

O caso de uso na Figura 7 é o que define como o agente de interface irá auxiliar o usuário do MOODLE para executar as funcionalidades do sistema. No momento que um usuário for realizar alguma atividade no AVA, o agente é acionado para explicar como essa funcionalidade opera. No entanto, o usuário também poderá perguntar diretamente ao agente sobre uma determinada funcionalidade que tenha dúvida.

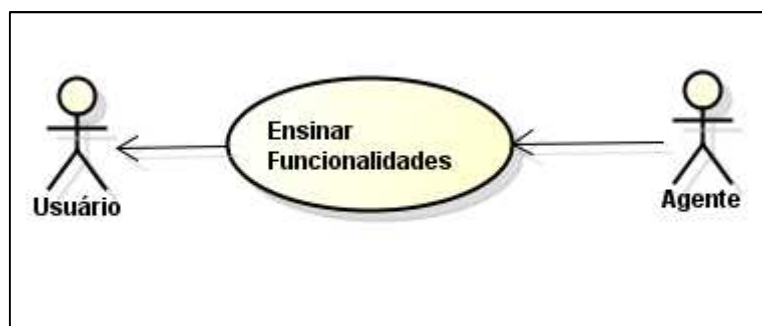


Figura 7 – Diagrama do Caso de Uso Ensinar Funcionalidades
Fonte: Autoria Própria

A Tabela 8 apresenta a descrição do caso de uso Ensinar funcionalidades e os passos que devem ser seguidos para executá-lo.

Tabela 8 - Caso de Uso Ensinar Funcionalidades

Nome	UC006 – Ensinar Funcionalidades
Descrição	Este caso de uso é responsável ensinar ao usuário como operar uma determinada funcionalidade disponível no MOODLE.
Ator Principal	Agente
Ator de Bastidor	Usuário
Pré Condições	O agente deve estar ativo.
Fluxo Básico	1. O usuário pergunta ao agente sobre a Funcionalidade X. 2. O agente informar ao aluno o passo a passo de como executar a Funcionalidade X.
Regras de Negócio	R002, R003 e R008

Fonte: Autoria Própria

Após realizar o levantamento dos requisitos e a formalização dos mesmos através dos casos de uso, é possível desenvolver o diagrama de classes para definir a estrutura do projeto. A próxima seção apresenta a estrutura do MOODLE gerada a partir dos casos de uso.

4.3 ESTRUTURA DO MOODLE

A fim de especificar como a estrutura do sistema deve ser construída para satisfazer as funcionalidades descritas nos casos de uso, este capítulo se destina a apresentar a estrutura do MOODLE para o desenvolvimento de uma nova funcionalidade, neste caso o agente de interface.

Para criar uma nova funcionalidade, denominada pelo MOODLE por módulo, é preciso criar um diretório com o nome do módulo e este diretório deve ser subdividido nos seguintes subdiretórios:

- *db*: este diretório contém os *scripts* de criação das tabelas que serão utilizadas pelo módulo;
- *lang*: diretório que contém os arquivos com os textos que serão apresentados pelo módulo em diferentes idiomas;
- *pix*: neste diretório ficam os ícones usados pelo módulo;
- *resource*: neste diretório ficam os recursos que poderão ser utilizados pelo módulo.

Além destes diretórios padrões do MOODLE, um módulo também deve conter alguns arquivos definidos pela estrutura do MOODLE:

- *index.php*: página inicial do módulo;
- *view.php*: página de visualização de um módulo;
- *mod_form.php*: página de cadastro do módulo;
- *lib.php*: contém as funções que serão usados pelo módulo, incluindo as interações com o banco de dados do MOODLE;
- *version.php*: contém as informações de versão do módulo.

Seguindo esses critérios, um diretório com o nome do agente de interface foi criado como um módulo seguindo os padrões definidos na documentação do MOODLE (MOODLEDocs).

Neste caso os arquivos .php gerados serão considerados como classes. A partir da definição dos arquivos do sistema necessário para a criação do agente é possível construir a estimativa do tempo necessário para a construção do agente de interface. O Apêndice A estima o tempo para o projeto com base nos casos de uso e classes definidos para o projeto.

4.4 MODELO ENTIDADE RELACIONAMENTO

Segundo Sommerville (2007), o modelo entidade relacionamento é a definição de forma lógica dos dados processados pelo sistema. Essa técnica mostra as entidades de dados, os atributos e os relacionamentos existente entre elas.

Para tratar dados específicos do agente de interface dentro do MOODLE, algumas tabelas foram modeladas e dispostas na Figura 8. Todas as tabelas criadas para uso exclusivo do agente de interface possuem o prefixo *mdl_gloob*, sendo o *mdl* prefixo da base de dados que representa o MOODLE e *gloob* o nome dado ao agente de interface. Desta forma as tabelas do agente de interface são diferenciadas das demais.

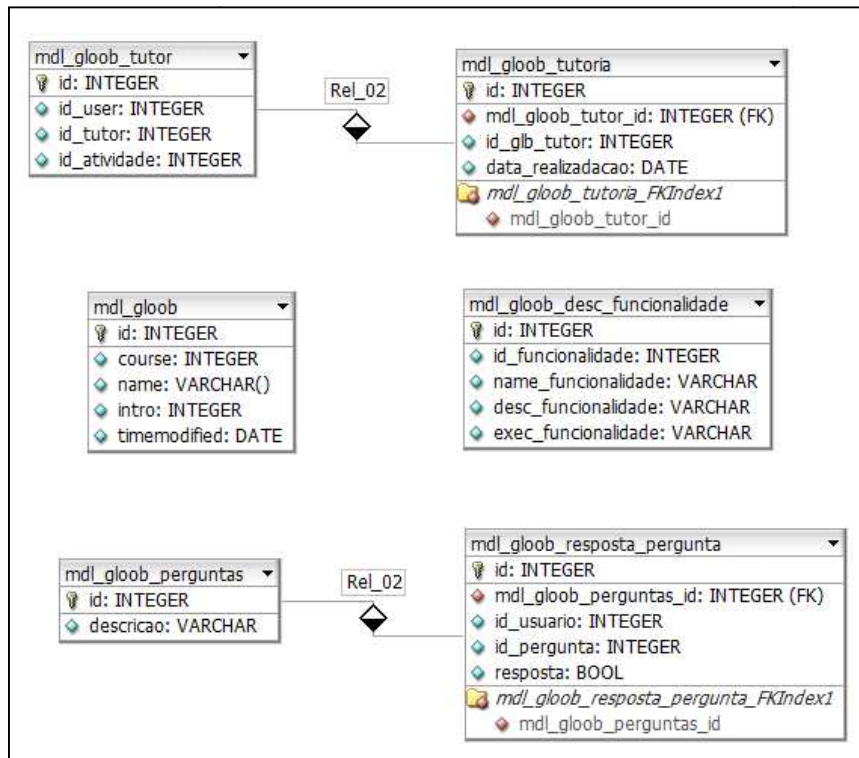


Figura 8 – Modelo Entidade Relacionamento Agente de Interface
Fonte: Autoria Própria

Para incluir dados relacionados à inclusão do agente de interface no MOODLE, a tabela *mdl_gloob* foi criada. Esta é uma tabela exigida pela estrutura do MOODLE para que seja possível adicionar funcionalidade ao sistema. Nesta tabela é armazenado o curso que o agente pertence, o nome que foi dado para ele no momento de sua criação, uma pequena introdução sobre o agente de interface e a data em que o mesmo foi criado.

A *mdl_gloob_desc_funcionalidade* é responsável por armazenar os dados das funcionalidades do MOODLE. A tabela contém o nome da funcionalidade, a sua descrição e a maneira pela qual o usuário pode executar tal funcionalidade. As perguntas que o agente apresenta no momento de sua conversação com o usuário, está armazenada em *mdl_gloob_perguntas*.

Os dados pertinentes à interação entre os alunos de cada curso, a fim de que os mesmos se auxiliem, estão persistidos nas tabelas *mdl_gloob_tutor* e *mdl_gloob_tutoria*. A *mdl_gloob_tutor* contém a relação entre um aluno que necessita de auxílio e o aluno que o agente escolheu para auxiliá-lo. A tabela *mdl_gloob_tutoria* armazena a

conversação desses usuários, se esta ocorrer, dentro de alguma forma de comunicação que o MOODLE oferece.

É importante ressaltar que o agente de interface ainda faz uso de tabelas próprias do MOODLE. Tais tabelas são apresentadas a seguir:

- *mdl_users*: armazena os dados dos usuários;
- *mdl_event*: contém os eventos que são criados no MOODLE;
- *mdl_course*: guarda os dados sobre os cursos;
- *mdl_modules*: registra as informações sobre os módulos do MOODLE.

Essas tabelas foram utilizadas para que o agente de interface conhecesse o ambiente no qual está localizado. Desta forma, o agente de interface obtém conhecimento inicial sobre o *software* no qual está inserido e o usuário que está manipulando o sistema.

Após definir o escopo do projeto foi preciso estimar o tempo necessário para a construção do agente de interface. O Apêndice A contém a descrição do método utilizado para estimar o esforço aplicado no desenvolvimento do agente de interface. O capítulo a seguir apresenta os resultados obtidos com o desenvolvimento deste trabalho.

5 GLOOB – O AGENTE DE INTERFACE

Este capítulo é destinado a apresentar o agente de interface desenvolvido neste trabalho. Uma das fases do projeto consistiu em definir a personagem para representação do agente de interface no ambiente MOODLE. Sabendo que o público alvo que irá utilizar o agente de interface serão alunos da UTFPR e seus professores, deve utilizar um personagem que se identificasse com este grupo além de ser compatível com a função do MOODLE que é o aprendizado. Portanto, a imagem de uma professora foi escolhida para personificar o agente de interface do MOODLE, a qual é apresentada pela Figura 9.

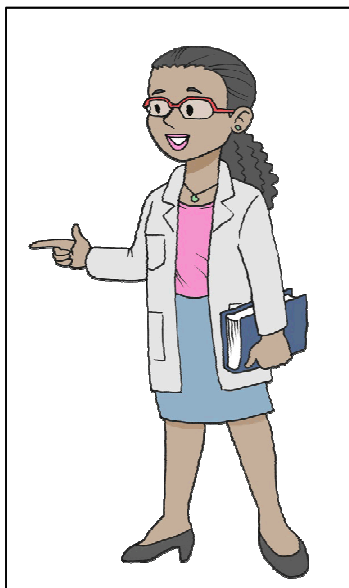


Figura 9 - Personagem Agente de Interface
Fonte: Caderno de Professora (2013)

Outro requisito importante é a comunicação do agente com o usuário. Esta deve ser realizada de forma dinâmica e para isso foi escolhida a biblioteca jQuery. Optando por uma forma simples, porém funcional, foram escolhidos balões de mensagem conforme a Figura 10 para realizar a comunicação do agente. Os balões de mensagens têm por função apresentar mensagens rápidas e claras ao usuário.



Figura 10 - Balão de Mensagens
Fonte: Autoria Própria

Com os balões é possível apresentar ao usuário mensagens em linguagem natural, o que torna a fácil à aceção das mensagens pelo usuário. Além dos balões, estão disponíveis botões (ver Figura 11) que permitem ao usuário interagir com o agente. O botão de “Próximo” serve para seguir com as informações que o agente está apresentando em seu balão de mensagem e o botão “Anterior” permite que o aluno retorne a uma mensagem já apresentada. Enquanto que os botões de “Sim” ou “Não” são para responder as perguntas do agente de interface.

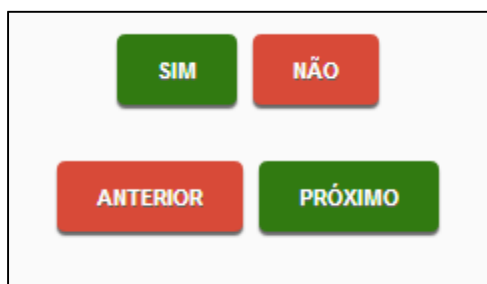


Figura 11 – Botões do Agente de Interface
Fonte: Autoria Própria

Durante a fase de projeto e desenvolvimento do agente as possíveis ações do personagem foram definidas a fim de que o usuário não criasse expectativas exageradas em torno do agente de interface. Isto é, o usuário não irá criar expectativas elevadas quanto ao grau de inteligência e isto não acarretará em desconfianças quanto à competência do agente de interface (MORAES, 2005), já que estas foram bem definidas no escopo do projeto.

Para a utilização do agente de interface é necessário implantá-lo ao MOODLE. Para isto, é preciso inserir a pasta do agente de interface dentro do diretório *mod* do MOODLE. Na sequência, é necessário aplicar o *scripts mysql.sql* do agente de interface

na base de dados do MOODLE. Após estes procedimentos, o agente de interface Gloop estará disponível para uso em qualquer curso do ambiente.

Desta forma, o professor poderá incluir o agente de interface na sua disciplina através da inclusão de uma atividade. Ao escolher o módulo Gloop, o professor será encaminhado para a página de criação do agente de interface. Nesta página deve ser informado o nome desejado para o agente de interface (ver Figura 12). Ainda é possível informar o grupo para o qual o agente de interface irá aparecer selecionando as opções disponíveis na combo *Group mode*. O campo *visible* indica se o agente ficará visível para os membros do curso. E o campo *ID number* é responsável por armazenar um número para a identificação do agente de interface pelo MOODLE.

Adding a new Gloop

General

Nome do agente*

Common module settings

Group mode

Visible

ID number

There are required fields in this form marked *

Figura 12 - Acrescentando um novo Agente de Interface
Fonte: Aatoria Própria

Após este procedimento o agente de interface estará disponível para qualquer integrante do curso. Para acessar o agente, o usuário deve clicar no *link* destinado ao agente de interface na página inicial do curso e o mesmo será redirecionado à página do agente de interface. O agente de interface dará boas vindas ao usuário, conforme ilustra a Figura 13.

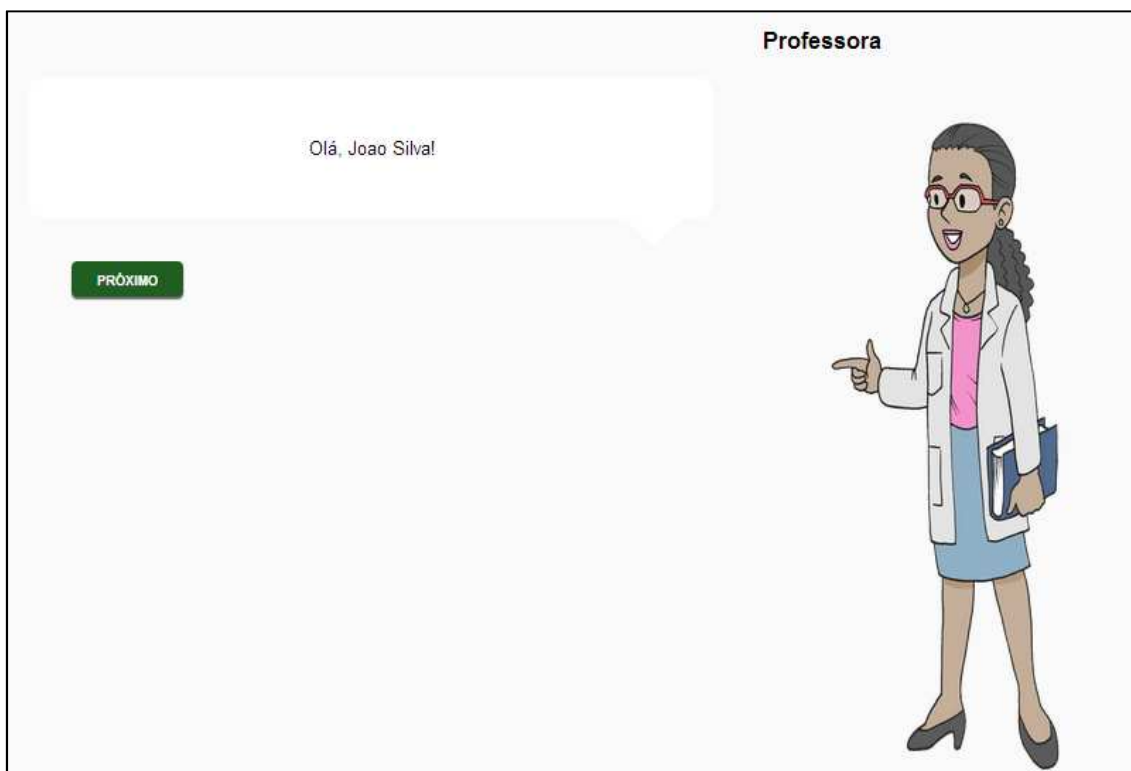


Figura 13 - Tela de Boas Vindas do Agente de Interface
Fonte: Autoria Própria

Em seguida, o usuário poderá manter a comunicação com o agente por meio dos botões. O agente poderá informar ao usuário os eventos disponíveis para ele dentro do curso, além de informar e descrever as funcionalidades existentes no MOODLE (ver Figura 14).

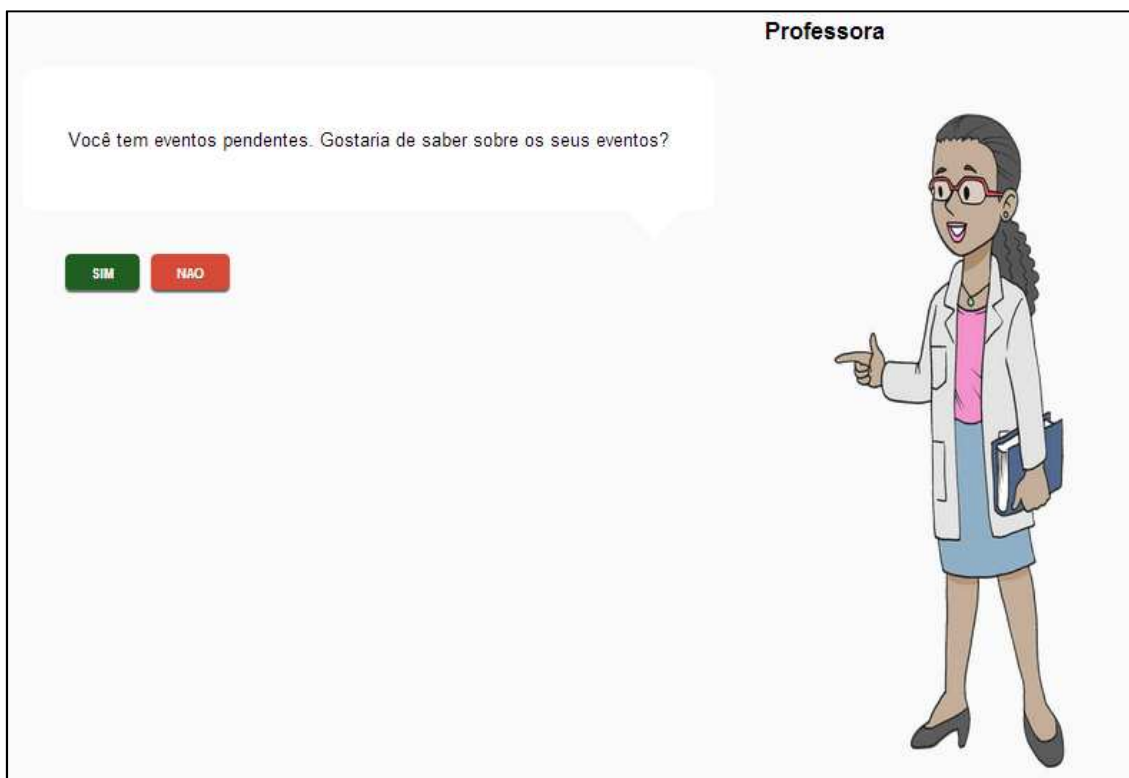


Figura 14 - Aviso de Eventos no Agente de Interface
Fonte: Autoria Própria

O agente também sugere ao usuário que realize alguma das atividades propostas no curso juntamente com um colega de disciplina. Podendo o usuário aceitar ou rejeitar a sugestão do agente de interface. Além disto, o agente apresenta ao usuário as funcionalidades presentes do MOODLE conforme a Figura 15.



Figura 15 - Informação sobre funcionalidade do MOODLE
Fonte: Autoria Própria

O agente informa ao usuário como ele pode executar a funcionalidade presente no MOODLE. No exemplo da Figura 15, o agente informa ao usuário que o professor do curso precisa criar um chat. Na sequência, o usuário, por meio do link disponibilizado na página principal do curso, pode acessar o chat e interagir com os demais participantes do curso, como pode ser observado na Figura 16.

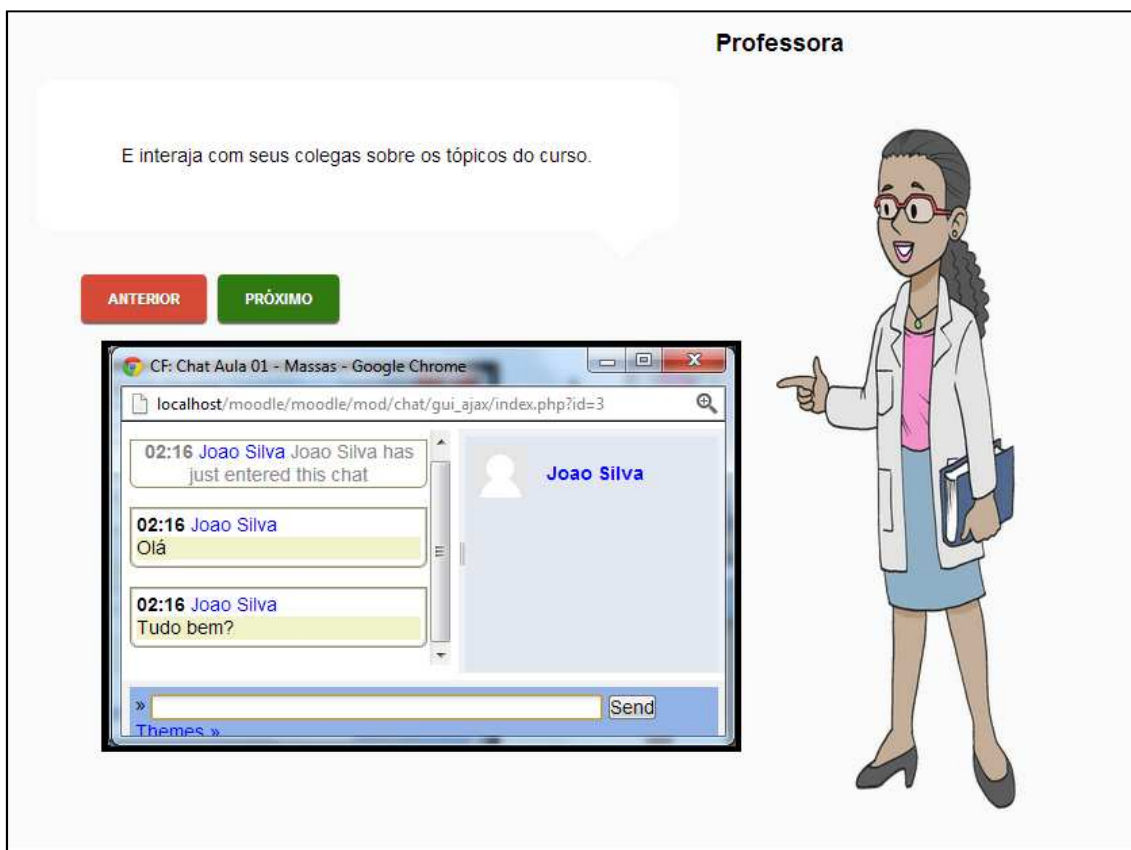


Figura 16 - Funcionalidade do MOODLE no Agente de Interface
Fonte: Autoria Própria

O capítulo a seguir apresenta os procedimentos de testes aplicados ao agente de interface deste trabalho.

6 PROCEDIMENTOS DE TESTE E VALIDAÇÃO

A fim de certificar se o agente de interface desenvolvido atende às especificações e apresenta as funcionalidades esperadas é criado um procedimento para validar se o produto final está de acordo com a especificação realizada na análise do projeto. Isto é, se o agente está apresentando de maneira adequada os requisitos funcionais e não funcionais especificados no início do projeto (SOMMERVILLE, 2007).

Para estabelecer o quanto o agente de interface está adequado à sua especificação, foram realizados testes de *software*. Tais testes examinam os dados de saída que são gerados após os dados de testes serem processados pelo agente. Esse processo exercita o agente, podendo assim descobrir defeitos ou inadequações no agente por meio do exame das saídas do mesmo.

De acordo com Sommerville (2007), existem dois tipos distintos de testes que podem ser aplicados. Um deles é o teste de validação que tem por finalidade mostrar se o *software* atende aos requisitos. O outro é o teste de defeitos, o qual é destinado a revelar defeitos no sistema em vez de simular o seu uso operacional.

No contexto do agente de interface para o MOODLE são aplicados testes do tipo validação, o qual vai identificar se o agente apresenta todas as funcionalidades descritas pelos requisitos. Para que isso seja concretizado, o Apêndice B apresenta os casos de testes que se aplicam ao agente. Cada caso de uso descrito na Seção 4.2 foi testado individualmente pela equipe de desenvolvimento do projeto. Desta forma, foi possível verificar cada funcionalidade específica isoladamente o que demonstra o quanto cada uma delas é operacional (PRESSMAN, 2011).

O primeiro caso de teste executado serviu para verificar se o agente de interface está apto a se comunicar com o usuário de maneira eficaz. O agente apresentou as suas mensagens por meio de balões de mensagens. Além disto, o agente ofertou ao usuário botões pelos quais o usuário pode responder as perguntas do agente ou simplesmente seguir para a próxima instrução.

O segundo caso de teste conferiu se o agente de interface é capaz de verificar os eventos pertinentes ao usuário. Durante a aplicação deste caso de teste foi possível verificar que o agente de interface informa ao usuário os últimos eventos criados dos quais ele deve participar.

No terceiro caso de teste é verificado se o agente de interface é capaz de selecionar um tutor para auxiliar o usuário na execução de suas atividades. Ao informar ao usuário que o mesmo possui eventos, o agente de interface oferece ao usuário um tutor para auxiliá-lo na execução destes eventos. O usuário pode ou não aceitar o tutor escolhido. Em caso de aceite, o agente informa os dados para que o usuário possa entrar em contato com o seu tutor. Caso o usuário recuse o tutor, o agente de interface seleciona outro tutor para o usuário.

O quarto caso de teste é responsável por verificar se o agente de interface está informando ao usuário as funcionalidades do MOODLE. Após informar ao usuário sobre os eventos, o agente de interface começa a sua apresentação sobre as funcionalidades que o MOODLE disponibiliza e ainda descreve como estas funcionalidades podem ser executadas, cobrindo o quinto caso de teste.

O próximo capítulo apresenta as conclusões e possíveis trabalhos futuros que podem ser aplicados a este trabalho.

7 CONCLUSÃO

Atualmente os agentes de *software* são amplamente utilizados e envolvem diversas subáreas da Ciência da Computação para a sua construção. Isto permite o aperfeiçoamento constante destes agentes, sendo os agentes de interface provenientes dessas atualizações. Esta tipologia de agente é aplicada na interação homem-máquina a fim de aprimorar a interface de diversos sistemas. Este método tenta alcançar os interesses dos seres humanos, sem que este indivíduo necessite se adequar à interface imposta pela aplicação.

O agente de interface está presente para identificar as necessidades do usuário e moldar ou apresentar o sistema da forma mais amigável para o usuário. Ele procura superar as limitações pertinentes das interfaces tradicionais e proporcionar ajuda personalizada. Um agente de interface ainda atua como guia automatizado de execução de tarefas complexas ou repetitivas. Com a necessidade de tornar os AVAs mais atrativos para os usuários, um agente de interface foi desenvolvido para ser integrado ao MOODLE, ambiente este que é utilizado como ferramenta de apoio ao ensino na UTFPR. Este trabalho foi indicado devido ao crescente uso desta aplicação entre alunos, professores e gestores (PAVEZI et al., 2011). Além disso, como forma de encontrar meios que facilitem o uso de sistemas comuns, sempre devem ser estudados e experimentados mecanismos que atinjam maior grau de utilização da ferramenta pelo usuário.

Foram estudados meios para construção do agente de interface de modo que este se aproximasse o máximo possível das necessidades do usuário e não perdesse a sua função de facilitador acarretando problemas ao usuário durante a execução de suas tarefas. O estudo dos métodos para a confecção deste agente foi de relevante importância para assegurar que o agente cumprisse com as suas funções sem interferir no desempenho do usuário.

7.1 TRABALHOS FUTUROS

A partir da construção deste agente de interface puderam-se perceber melhorias, as quais podem ser desenvolvidas a fim de agregar valor ao agente de interface. Uma das incorporações que seriam importantes para aprimorar o agente de interface é a aplicação de padrões de *design* e interação visando o aperfeiçoamento na relação homem-máquina, criando maior conforto ao usuário no momento de sua interação com o agente de interface. Como trabalho futuro pode-se ainda criar um modelo no qual o agente possa construir suas frases para conversação com usuário de forma dinâmica a partir de sua base de conhecimentos sobre o usuário, dado que atualmente ele se utiliza de frases prontas para construir um diálogo com o usuário do sistema. Além disso, é possível agregar ao MOODLE um novo *thema*, recurso implícito do sistema. A criação deste *thema* possibilitará a sobreposição do agente de interface sobre todo o AVA de modo que o agente possa auxiliar o usuário com informações de todo o sistema.

REFERÊNCIAS BIBLIOGRÁFICAS

ALVES, Lynn; BARROS, Daniela; OKADA, Alexandra. **MOODLE: Estratégias Pedagógicas e Estudos de Caso**. Salvador: EDUNEB, 2009.

BATES, J. **The Nature of Characters in Interactive Worlds and The OZ Project**. *Technical Report*, Pittsburgh (CMU-CS-92-200), 1992.

BRAGA, Bruno T. da R; PEREIRA, José L. A. **Agentes inteligentes – conceitos, características e aplicações**. Universidade da Amazônia, p.50, 2001.

CUNHA, Fabricio O.; SILVA, Julia M. C. **Análise das Dimensões Afetivas do Tutor em Turmas de EaD no Ambiente Virtual MOODLE**. XX Simpósio Brasileiro de Informática na Educação, 2009.

CRISTINA, Michele. **Caderno de Professora**. Disponível em: <<http://cadernodeprofessora.blogspot.com.br/2010/07/sei-que-vale-pena-ser-professora.html>>. Acessado em: 12 de fevereiro de 2013.

DANTAS, Teresa C. B et al. **AprendEAD: Ambiente para Educação à Distância Apoiado em Agentes**. Cadernos do IME: Série Informática, v. 23, 2007.

ELLIOTT, C.; BRZEZINSKY, J. **Autonomous Agents as Synthetic Characters**. AI Magazine. New York, NY: AAAI Press. Summer, p. 13-30, 1998

FERREIRA, Tânia; MOZZAQUATRO, Patrícia M. **Spike: um agente conversacional integrado ao Ambiente Virtual de Aprendizagem MOODLE**. Universidade de Cruz Alta, 2010.

FILHO, Jaime B. S. S. **Agentes de Software para o Ambiente Virtual de Aprendizagem MOODLE: Facilidades de Gerência**. *Revista de Sistemas de Informação*, Belém, v. 4 N° 2, 2007.

Free Software Foundation. Disponível em: <<http://www.gnu.org/copyleft/gpl.html>>. Acessado em: 6 de abril de 2012.

KARNER, Gustav. **Resource estimation for objectory projects**. *Objectory Systems*, 1993. Disponível em <<http://www.bfpug.com.br/artigos.htm>>. Acessado em: 11 de abril de 2013.

LACHI, Ricardo L. **Chapa: Um Agente de Interface para Ferramentas de Bate-papo em Ambientes de Ensino à Distância na Web**. Dissertação apresentada ao Instituto de Computação, UNICAMP. Campinas: (s.n.), 2003.

LAWINSCKY, Fabiana M.; HAGUENAUER, Cristina. **Análise das ferramentas da plataforma MOODLE do LATEC/UFRJ segundo a abordagem sistêmica relacional de interação**. XVII Congresso Internacional de Educação a Distância. 2011.

MAES, Pattie. **Agents that reduce work and information overload.** *Communications of the ACM*, v. 37, nº 7, 1994.

MOODLE. Disponível em:< <http://MOODLE.org/> >. Acessado em: 22 de abril de 2012.

MOODLE Docs. Disponível em <http://docs.MOODLE.org/24/en/Main_page>. Acessado em: 27 de dezembro de 2012.

MORAES, Márcia C. **Agentes Animados de Interface.** XXV Congresso da Sociedade Brasileira de Computação, 2005.

MySQL. Disponível em:< <http://www.mysql.com/>>. Acessado em: 28 de abril de 2012.

Microsoft. Disponível em:<<http://www.microsoft.com/products/msagent/main.aspx>>. Acessado em: 20 de março de 2013.

NWANA, Hyacinth S. **Software Agents: An Overview.** *Knowledge Engineering Review*, v. 11, nº 3, 1996.

PAVEZI, Adriana M. et al. **O Uso Das Ferramentas Do Ambiente Virtual De Aprendizagem Pelos Acadêmicos Dos Cursos De Administração E Processos Gerenciais Do Nead-Cesumar XVII.** Congresso Internacional de Educação a Distância, 2011.

PRESSMAN, Roger S. **Engenharia de software: uma abordagem profissional.** 7. ed. Porto Alegre, RS: AMGH, 2011. 780 p..

RABELLO, Roberto dos S.; REATEGUI, Eliseo; COSTI, Julio C. G. **Um Agente de Interface Aplicado ao Ensino de Equações de Primeiro Grau.** *CINTED-UFRGS*, Porto Alegre, v. 5, nº 2, 2007.

RIBEIRO, João P. A; REATEGUI, Eliseo; BOFF, Elisa. **Integrando um Agente Pedagógico para Recomendação de Tutores a um Sistema de Gerência de Cursos.** Universidade de Caxias do Sul, p.10, 2007.

RUSSEL, Sturart J.; NORVIG, Peter. **Inteligência Artificial.** Editora Campus - 2º ed. Rio de Janeiro, p. 10-21, 2004.

SOMMERVILLE, Ian. **Engenharia de software.** 8. ed. São Paulo: Pearson Addison-Wesley, 2007. xiv, 552 p

VAHL, J.; OEIRAS J. ; ROCHA H. **Uso de agentes de interface para adequação de Bate-papos ao contexto de educação a distância.** IHC 2002.

VIEIRA, Everton L. **Uso do Conceito de Passos Obrigatórios para Aprimorar o Processo de Contagem do Método “Pontos de Caso de Uso”.** UFSC, Florianópolis, Junho de 2007. Disponível em:

<<http://www.bfpug.com.br/Artigos/UCP/Everton%20Luiz%20Vieira.pdf>>. Acessado em: 27 de junho de 2012.

APÊNDICE A – Estimativa de Tempo para a construção do agente de interface

Com base na definição dos requisitos é possível fazer a estimativa precisa do esforço necessário para o desenvolvimento do agente de interface. Essa estimativa fornece uma medida do esforço humano exigido (pessoa/mês) e a duração cronológica de todo o projeto de desenvolvimento do agente de interface.

Para estimar o esforço na construção do agente de interface, uma técnica é aplicada sobre os casos de uso gerados durante a análise deste projeto. Essa técnica exige uma declaração definida do escopo, o qual é obtido através da formalização dos requisitos em casos de uso (PRESSMAN, 1995).

O primeiro passo é estimar a complexidade de cada caso de uso, seja por meio de transações a serem realizadas ou pelo número de classes que serão geradas para a construção de cada caso de uso. Neste trabalho, a complexidade é definida pelo número de classes que cada caso de uso deve gerar para ser desenvolvido. Sendo assim, a classificação é dada da seguinte forma seguindo o modelo proposto em Vieira (2007):

- Caso de uso simples (5 pontos): cria menos do que 5 classes ou tabelas geradas na análise do projeto;
- Caso de uso médio (10 pontos): gera entre 5 e 10 classes ou tabelas; e
- Caso de uso complexo (15 pontos): possui mais do que 10 classes ou tabelas.

Analisando a Tabela 9, a qual apresenta a Complexidade dos Casos de Uso (CCU) para a construção do agente, é possível perceber que do total de seis casos de uso nenhum foi considerado complexo. No entanto, quatro deles foram classificados com complexidade média e dois com complexidade simples. A soma dos produtos do número de casos de uso de cada tipo pelo seu peso fornece a CCU.

Tabela 9 - Complexidade de Casos de Uso

Complexidade	Quantidade de Casos de Uso	Peso	Resultado
Simple	4	5	20
Médio	1	10	0
Complexo	0	15	0
Total CCU			20

Fonte: Autoria Própria

Após a classificação dos casos de uso é necessário medir a complexidade dos atores. A Tabela 10 mostra que o projeto possui apenas dois atores envolvidos. Sendo um considerado simples e o outro complexo. Os atores definidos como simples recebem um ponto enquanto que os atores complexos recebem três pontos (VIEIRA, 2007). A soma dos produtos do número de atores de cada tipo pelo respectivo peso fornece a Complexidade dos Atores (CA).

Tabela 10 - Complexidade dos Atores

Complexidade	Quantidade Atores	Peso	Resultado
Simple	1	1	1
Médio	0	2	0
Complexo	1	3	3
Total CA			4

Fonte: Autoria Própria

Com os pontos gerados pelos casos de uso e atores é possível obter os Pontos de Casos de Uso Não Ajustados (PCUNA) por meio da Equação (1). No total do trabalho, o valor do PCUNA é 24.

$$PCUNA = CCU + CA, \quad (1)$$

onde: CCU são os pontos gerados pelos casos de uso e CA é o total de pontos gerados pelos atores.

Depois da avaliação dos casos de uso e dos atores é necessário avaliar os fatores técnicos e ambientais do projeto. Os fatores técnicos são relativos aos requisitos

funcionais do projeto. Enquanto, os fatores ambientais representam os requisitos não funcionais, a experiência e motivação da equipe (VIEIRA, 2007).

Em cada fator é determinado um valor entre zero e cinco, dependendo da influência que o mesmo tenha sobre o projeto. A Tabela 11 apresenta o significado de cada valor de influência.

Tabela 11 - Influência dos Fatores Técnicos e Ambientais

Influência	Significado
0	Nenhuma
1	Irrelevante
2	Influência Mínima
3	Influência Moderada
4	Influência Significativa
5	Essencial

Fonte: Autoria Própria

A lista com o peso e a influência de cada fator avaliado para o desenvolvimento da aplicação considerada está descrito na Tabela 12. Esses pesos refletem os valores atribuídos por Vieira (2007). O valor técnico FTécnico representa o quanto cada fator técnico é importante para o projeto. Isto é, corresponde a multiplicação do peso de cada fator técnico com a influência que o mesmo exerce sobre o processo de desenvolvimento do projeto. É importante ressaltar que os fatores técnicos T5 e T12 não são aplicáveis à construção do agente de interface proposto neste trabalho.

Tabela 12 - Fatores Técnicos

Fator	Descrição	Peso	Influência	FTécnico
T1	Sistema distribuído	2	1	2
T2	Objetivos de <i>performance</i>	1	4	4
T3	Eficiência on-line	1	5	5
T4	Complexidade de processamento	1	5	5
T5	Código reusável em outras aplicações	1	0	0
T6	Facilidade de instalação	0,5	4	2
T7	Facilidade de uso	0,5	5	2,5
T8	Portabilidade	2	3	6
T9	Facilidade de alterações	1	3	3
T10	Concorrência	1	5	5
T11	Segurança	1	5	5
T12	Acesso direto a terceiros	1	0	0
T13	Necessidade de facilidades especiais de treinamento para usuários	1	2	2
FTécnico				41,5

Fonte: Autoria Própria

A partir dos valores apresentados na Tabela 12, é possível definir o Fator de Complexidade Técnica (FCT) do projeto em relação aos seus casos de uso através da Equação (2).

$$FCT = 0,6 + (0,01 * FTécnico) \quad (2)$$

onde: FTécnico é obtido pelo somatório dos pesos de cada fator técnico multiplicados pela sua influência.

Aplicando os valores na Equação (2), tem-se o fator de complexidade técnica do projeto de 1,015.

A Tabela 13 apresenta uma lista do peso e influência de cada fator ambiental avaliado para o desenvolvimento do projeto em questão. O valor FAmbiental representa o quanto cada fator ambiental é importante para o projeto.

Tabela 13 - Fatores Ambientais

Fator	Descrição	Peso	Influência	FAmbiental
F1	Familiaridade da equipe com RUP	1,5	2	3
F2	Experiência da equipe	0,5	2	1
F3	Experiência da equipe em OO	1	3	3
F4	Capacidade dos analistas da equipe	0,5	2	1
F5	Motivação	1	3	3
F6	Estabilidade dos requisitos	2	3	6
F7	Estagiários ou funcionários em tempo parcial	-1	5	-5
F8	Domínio da tecnologia e configuração do ambiente	-1,5	4	-6
FAmbiental				6

Fonte: Autoria Própria

O Fator Ambiental é calculado de acordo com a Equação (3).

$$FA = 1.4 + (-0,03 * FAmbiente) \quad (3)$$

onde: FAmbiente é obtido pela somatória dos pesos de cada fator ambiental multiplicados pela sua influência correspondente.

Desta forma, é possível verificar que o valor do fator ambiental para a construção do agente de interface equivale a 1,22.

Com os valores dos fatores ambiental e técnico, é possível estabelecer os Pontos de Casos de Uso (PCU) gerados pelo projeto de acordo com a Equação (4).

$$PCU = PCUNA * FCT * FA, \quad (4)$$

onde: PCUNA são os pontos de casos de uso não ajustados, FTC (ver Equação (1)) é o fator de complexidade técnica, e FA é o fator ambiental (ver Equação (3)).

Após aplicação dos valores correspondentes na Equação (4), é possível verificar que o valor de PCU para este projeto equivale a 29,72 pontos.

Na sequência é necessário dividir os pontos de casos de uso pelo número de pessoas que irão trabalhar nesse projeto. Como somente uma pessoa pertence à equipe de desenvolvimento do agente de interface, a esta pessoa é atribuído 29,72 pontos para desenvolver.

Segundo Karner (KARNER, 1993), o tempo necessário para o desenvolvimento de cada ponto de uso é de 20 horas de trabalho. Sendo assim o total de horas para todo o desenvolvimento de todo o trabalho é de 594,4 horas.

A pessoa destinada ao desenvolvimento, implantação e testes deve construir todo o projeto em aproximadamente seis meses e meio, por esta estar disposta a utilizar cerca de três horas diárias para esse desenvolvimento o projeto agente de interface para o ambiente MOODLE.

APÊNDICE B – Casos de teste

	Criado por	Melina Deraldo dos Santos
	Data:	28/06/2012
	Data da última revisão	28/06/2012
	Status:	
	Número de Passos:	19
Version 1.0		
Teste de Casos de Uso		

Pré Condições	Não se Aplica
----------------------	---------------

UC001 - Caso de Uso Conversar com usuário				
Test Case	Ações	Resultados Esperados	Resultados:	Observações
1.1.1	1 – Entrar na página inicial do MOODLE. 2 – Ativar agente de interface.	O agente apresenta balão de conversa com a seguinte mensagem: "Olá Usuário! Existe X novos Eventos".		
1.1.1	1 – Usuário clica em OK. 2 – O Usuário ativa modo de conversação	O agente apresenta uma caixa de diálogo.		
1.1.2	1 – Usuário digita texto envolvendo funcionalidades do MOODLE.	O agente responde sobre a funcionalidade perguntada.		

UC002 - Caso de Uso Verificar Eventos				
Casos de Teste	Ações	Resultados Esperados	Resultados	Observações
2.1.0	1 – O usuário entra na página inicial do MOODLE 2 – O agente apresenta balão de apresentação 3 – O usuário pede para ver os eventos do MOODLE	O agente apresenta mensagem de confirmação: "Deseja ver todos os últimos eventos".		
2.1.1	1 – Usuário clica em OK.	O agente apresenta uma lista com todos os eventos criados desde o último acesso do usuário.		
2.1.2	1 – O usuário seleciona um dos eventos apresentados pelo agente.	O agente apresenta detalhes sobre o evento selecionado.		
2.1.3	1 – Usuário clica no link da página do evento.	O agente direciona o usuário para a página do evento.		

UC003 - Caso de Uso Conhecer Tutor				
Casos de Teste	Ações	Resultados Esperados	Resultados:	Observações
3.1.0	1 – O usuário entra na página inicial do MOODLE 2 – O agente apresenta balão de apresentação 3 – O usuário pede para conhecer o Tutor 4 – O usuário entra em uma determinada disciplina 5 – O usuário pede ao agente para conhecer o tutor	O agente apresenta balão de conversa com a seguinte mensagem: O seu Tutor é selecionado.		
3.1.1	1 – Usuário clica em OK.	O agente informa o nome do Tutor selecionado para o aluno. O agente envia uma notificação ao aluno selecionado como Tutor.		
3.1.2	1 – O usuário clica em OK	O agente sugere para o usuário utilizar o bate-papo do MOODLE e direciona o usuário para a página do bate-papo.		
3.1.3	1 – O usuário clica em OK	O agente sugere para o usuário utilizar o bate-papo do MOODLE e direciona o usuário para a página do bate-papo.		

UC004 - Informar funcionalidades				
Caso de Teste	Ações	Resultados Esperados	Resultados	Observações
5.1.2	1 – O usuário entra na página inicial do MOODLE 2 – O usuário ativa uma funcionalidade.	O agente pergunta ao usuário se ele quer mais informações sobre a funcionalidade.		
5.1.1	1 – O usuário clica em OK.	O agente explica sobre a funcionalidade.		
5.2.1	1 – O usuário entra na página inicial do MOODLE 2 – O usuário pergunta ao agente sobre as funcionalidades do MOODLE.	O agente explica sobre a funcionalidade		

UC005 - Ensinar funcionalidades				
Caso de Teste	Ações	Resultados Esperados	Resultado	Observações
6.1.0	1 – O usuário entra na página inicial do MOODLE 2 – O pergunta ao agente sobre uma funcionalidade. 3 – O agente informa sobre a funcionalidade 4 – O usuário pede para o agente o caminho de execução da funcionalidade 5 – O usuário pede ao agente para conhecer o tutor	O agente informa ao usuário o caminho para executar a funcionalidade		